

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО**

Факультет інформатики та обчислювальної техніки
(назва факультету, інституту)

Кафедра автоматизованих систем обробки інформації і управління
(назва кафедри)

"На правах рукопису"
УДК 004.023

«До захисту допущено»
Завідувач кафедри

(підпис) О.А.Павлов
(ініціали, прізвище)
“ ” 20 18 р.

**МАГІСТЕРСЬКА ДИСЕРТАЦІЯ
на здобуття ступеня магістра**

за спеціальністю 126 Інформаційні системи та технології
(код та назва спеціальності)

ОПП Інформаційні управляючі системи та технології
(код та назва спеціалізації)

на тему: Рекомендаційна інформаційна система на основі вподобань користувачів

Виконав: студент VI курсу групи ІС-72мп
(шифр групи)

Козулько Олександр Сергійович
(прізвище, ім'я, по батькові) _____ (підпис)

Науковий керівник к.т.н., доц. Попенко В.Д.
(посада, науковий ступінь, вчене звання, прізвище та ініціали) _____ (підпис)

Консультант к.т.н., доц. Жданова О.Г.
(науковий ступінь, вчене звання, прізвище, ініціали) _____ (підпис)

Рецензент _____
(посада, науковий ступінь, вчене звання, прізвище та ініціали) _____ (підпис)

Засвідчую, що у цій магістерській дисертації
немає запозичень з праць інших авторів без
відповідних посилань.

Студент _____
(підпис)

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. ІГОРЯ СІКОРСЬКОГО»**

Факультет інформатики та обчислювальної техніки
(повна назва)

Кафедра автоматизованих систем обробки інформації та управління
(повна назва)

Рівень вищої освіти другий (магістерський) за освітньо-професійною програмою

Спеціальність 126 Інформаційні системи та технології
(код і назва)

ОПП Інформаційні управляючі системи та технології
(код і назва)

ЗАТВЕРДЖУЮ
Завідувач кафедри

(підпис) О.А.Павлов
(ініціали, прізвище)
«__» грудня 2018 р.

ЗАВДАННЯ

на магістерську дисертацію студенту

Козулько Олександр Сергійовичу

(прізвище, ім'я, по батькові)

1. Тема дисертації Рекомендаційна інформаційна система на основі вподобань користувачів

науковий керівник дисертації Попенко Володимир Дмитрович, к.т.н. доц.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від “ 07 ” листопада 20 18 р. № 4112с

2. Строк подання студентом дисертації “ 3 ” грудня 20 18 р.

3. Об'єкт дослідження процес формування особистих рекомендацій на основі вподобань користувачів.

4. Перелік завдань, які потрібно розробити 1) виконати огляд існуючих методів та алгоритмів роботи рекомендаційних систем; 2) здійснити порівняльний аналіз різних методів та алгоритмів роботи рекомендаційних систем; 3) формалізувати задачу формування рекомендацій на основі гібридного підходу; 4) розробити прототип рекомендаційної системи книг, використовуючи вищезазначені підходи; 5) виконати аналіз отриманих результатів.

5. Орієнтовний перелік ілюстративного матеріалу 1) ER-діаграма;
2) Діаграма класів; 3) Діаграма послідовності; 4) Діаграма компонентів;
5) Діаграма розгортання; 6) Схема роботи алгоритмів

6. Орієнтовний перелік публікацій 1) Когулько О.С., Попенко В.Д. Надання

Всеукраїнська науково-практична конференція молодих вчених та студентів «Інформаційні системи та технології управління» (ІСТУ-2018) – м. Київ.: НТУУ «КПІ ім. Ігоря Сікорського», 29-30 грудня 2018 р. 2) Когулько О.С. Використання методів практична 29
колаборативної фільтрації для роботи рекомендаційної системи / Міжнародна науково-
конференція «Математичне та імітаційне моделювання систем» (МОДС-2018) – м. Київ., 25-
червня 2018 р. – С. 83-86

7. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

8. Дата видачі завдання “ 29 ” жовтня 20 18 р.

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Строк виконання етапів магістерської дисертації	Примітка
1	<i>Систематизація результатів огляду літератури</i>		
2	<i>Порівняльний аналіз існуючих методів розв'язання задачі</i>		
3	<i>Постановка та формалізація математичної моделі задачі</i>		
4	<i>Модифікація існуючих методів розв'язання задачі</i>		
5	<i>Розробка інформаційного та програмного забезпечення</i>		
7	<i>Проведення експериментальних досліджень розроблених алгоритмів</i>		
8	<i>Оформлення документації</i>		
9	<i>Подання роботи на попередній захист</i>		
10	<i>Подання роботи на основний захист</i>		

Студент

(підпис)

О.С. Когулько

(ініціали, прізвище)

Науковий керівник

(підпис)

В.Д. Попенко

(ініціали, прізвище)

РЕФЕРАТ

Магістерська дисертація: 94 с., 19 рис., 27 табл., 1 додаток , 36 джерел.

Актуальність. Сьогодні, одним з найбільш цінних ресурсів людства – є інформація. Вона займає одну з центральних ролей у процесі формування сучасного суспільства. З виникненням Інтернету – основним сховищем інформації стала саме Всесвітня Мережа. Але з часом, її об'єми почали зростати значно швидше, ніж обчислювальні можливості, для обробки даних. Кількість інформації в Інтернеті настільки велика, що людина просто не здатна знайти те, що їй дійсно потрібно.

У зв'язку з цим актуальною є розробка спеціальної системи, яка буде сама пропонувати користувачу деякі елементи, яка вона буде вважати доцільними. Така система буде формувати свої рекомендації на основі поведінки користувачів в минулому та їх вподобань. Це дозволить користувачам зекономити великий об'єм часу для пошуку необхідного контенту.

Зв'язок роботи з науковими програмами, планами, темами. Робота виконувалась на кафедрі автоматизованих систем обробки інформації та управління Національного технічного університету України «Київський політехнічний інститут ім. Ігоря Сікорського» в рамках теми «Ефективні методи колаборативної фільтрації, засновані на аналізі поведінки користувачів, відображеної в "big data"»

Мета дослідження – підвищення ефективності роботи рекомендаційної системи.

Для досягнення мети необхідно виконати наступні **завдання**:

- виконати огляд існуючих методів та алгоритмів роботи рекомендаційних систем;
- здійснити порівняльний аналіз різних методів та алгоритмів роботи рекомендаційних систем;
- формалізувати задачу формування рекомендацій на основі гібридного підходу;
- розробити ефективну модель формування рекомендацій на основі гібридного підходу;

- розробити прототип рекомендаційної системи книг, використовуючи вищезазначені підходи;
- виконати аналіз отриманих результатів.

Об’єкт дослідження – процес формування особистих рекомендацій на основі вподобань користувачів.

Предмет дослідження – методи та моделі формування особистих рекомендацій.

Методи дослідження, застосовані у даній роботі, базуються на методах інформаційного пошуку.

Наукова новизна одержаних результатів полягає у застосуванні гібридного підходу до фільтрації даних для формування особистих рекомендацій. Розроблено підхід, який поєднує в собі фільтрацію даних, на основі вмісту з колаборативною та демографічною фільтрацію.

Публікації. Матеріали роботи опубліковані в тезах міжнародної науково-практичній конференції «Математичне та імітаційне моделювання систем МОДС 2018» а також у рамках Всеукраїнська науково-практична конференція молодих вчених та студентів «Інформаційні системи та технології управління» (ІСТУ-2018).

РЕКОМЕНДАЦІЇ, КОЛАБОРАТИВНА ФІЛЬТРАЦІЯ, ФІЛЬТРАЦІЯ НА ОСНОВІ ВМІСТУ, РЕЛЕВАНТНІСТЬ, КЛАСИФІКАЦІЯ, МІРА СХОЖОСТІ, ПОШУК, РЕЙТИНГ.

ABSTRACT

Master's dissertation: 94 pages., 19 Fig., 37 tabl., 1 addition, 36 sources.

Topicality. Today, one of the most valuable resources of mankind is information. It occupies one of the central roles in the process of formation of modern society. With the advent of the Internet - the main storage of information has become the World Wide Web. But over time, its volumes began to grow much faster than computing capabilities for data processing. The amount of information on the Internet is so great that a person simply can not find what she really needs.

In this regard, the development of a special system is urgent, which will recommend to user some elements that she deems advisable. Such a system will formulate its recommendations based on the behavior of users in the past and their preferences. This will allow users to save a large amount of time to find the content they need.

Relationship of work with scientific programs, plans, themes. The work was carried out at the Department of Automated Systems for Information Processing and Management of the National Technical University of Ukraine "Kyiv Polytechnic Institute. Igor Sikorsky" within the theme «Effective methods of collaborative filtering based on the analysis of user behavior, reflected in "big data"»

Purpose of the study is to improve the effectiveness of the recommendation system.

To achieve the **purpose**, we need to accomplish following tasks:

- review the existing methods and algorithms of the recommendation systems;
- carry out a comparative analysis of various methods and algorithms of the recommendation systems;
- formalize the problem of forming recommendations based on the hybrid approach;
- develop an effective model for recommendation computing based on a hybrid approach;

- develop a prototype of the books recommendation system by using the aforementioned approaches;
- perform the analysis of the results.

The object of the research - the process of creating personal recommendations based on the preferences of users.

Subject of research - methods and models for the formation of personal recommendations.

The research methods used in this paper are based on the methods of information retrieval.

The scientific novelty of the results is applying a hybrid approach to data filtering for the formation of personal recommendations. An approach is developed that combines data filtering based on content with collaborative and demographic filtering.

Publications: the materials of the work are published in the theses of the international scientific and practical conference "Mathematical and imitation systems modelling MISIM 2018" as well as within the framework of the All-Ukrainian Scientific and Practical Conference of Young Scientists and Students "The actual problems of informatization of management decisions" (APIMD 2018).

RECOMMENDATIONS, COLLABORATIVE FILTERING, CONTENT-BASED FILTERING, RELEVANTITY, CLASSIFICATION, OPPORTUNITY, SEARCH, RATING

ЗМІСТ

ВСТУП	9
1 ПРОЕКТНІ РІШЕННЯ З РОЗРОБКИ РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ НА ОСНОВІ ВПОДОБАНЬ КОРИСТУВАЧІВ.....	10
1.1 ОПИС БІЗНЕС-ПРОЦЕСІВ	10
1.1.1 Опис процесу діяльності.....	10
1.1.2 Актори і функції.....	13
1.1.3 Структура бізнес-процесів	16
1.2 ОПИС ПОСТАНОВКИ ЗАДАЧІ	17
1.3 ОГЛЯД АНАЛОГІВ СИСТЕМИ, ЩО РОЗРОБЛЯЄТЬСЯ	19
1.4 РІШЕННЯ З ІНФОРМАЦІЙНОГО ЗАБЕЗПЕЧЕННЯ	20
Висновок до розділу	30
2 МОДЕЛІ ТА МЕТОДИ ФОРМУВАННЯ РЕКОМЕНДАЦІЙ	31
2.1 ЗМІСТОВНА ПОСТАНОВКА ЗАДАЧІ	31
2.2 МАТЕМАТИЧНА МОДЕЛЬ	32
2.3 ОГЛЯД МЕТОДІВ РОЗВ'ЯЗАННЯ.....	33
2.3.1 ФІЛЬТРАЦІЯ НА ОСНОВІ ВМІСТУ	33
2.3.2 КОЛАБОРАТИВНА ФІЛЬТРАЦІЯ	37
2.3.3 ГІБРИДНИЙ ПІДХІД.....	41
2.3.4.1 ВКЛЮЧЕННЯ ХАРАКТЕРИСТИК ЕЛЕМЕНТІВ В КОЛАБОРАТИВНУ ФІЛЬТРАЦІЮ.....	42
2.3.4.2 ВКЛЮЧЕННЯ ХАРАКТЕРИСТИК КОРИСТУВАЧІВ В ФІЛЬТРАЦІЮ ЗА ВМІСТОМ.....	42
2.3.4.3 ПОБУДОВА ЄДИНОЇ МОДЕЛІ НА ОСНОВІ КОЛАБОРАТИВНОЇ ФІЛЬТРАЦІЇ ТА ФІЛЬТРАЦІЇ ЗА ВМІСТОМ	42
2.4 МОДИФІКАЦІЯ МЕТОДУ РОЗВ'ЯЗАННЯ ЗАДАЧІ	43
2.5 АЛГОРИТМ ЗАСТОСУВАННЯ МОДИФІКОВАНОГО ПІДХОДУ	46
2.6 РЕЗУЛЬТАТИ ДОСЛІДЖЕНЬ ЕФЕКТИВНОСТІ МЕТОДУ	47
Висновок до розділу	51
3 ОПИС ПРОГРАМНОГО ТА ТЕХНІЧНОГО ЗАБЕЗПЕЧЕННЯ.....	52
3.1 ЗАСОБИ РОЗРОБКИ.....	52
3.2 АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	55
3.2.1 ДІАГРАМА КЛАСІВ	55
3.2.2 ДІАГРАМА ПОСЛІДОВНОСТІ.....	57
3.2.3 ДІАГРАМА КОМПОНЕНТІВ	59
3.3 ІНСТРУКЦІЯ КОРИСТУВАЧА	60
3.4 ОПИС ТЕХНІЧНОГО ЗАБЕЗПЕЧЕННЯ.....	64
Висновок до розділу	65
4 РОЗРОБКА СТАРТАП-ПРОЕКТУ	66

	8
4.1 Опис ідеї стартап-проекту	66
4.1.1 Аналіз потенційних техніко-економічних переваг ідеї	66
4.2 Технологічний аудит ідеї проекту	68
4.3 Аналіз ринкових можливостей запуску стартап-проекту	69
4.4 Розроблення ринкової стратегії проекту	77
4.5 Розроблення маркетингової програми стартап-проекту	80
Висновок до розділу	82
ВИСНОВКИ	83
ПЕРЕЛІК ПОСИЛАНЬ	84
ДОДАТОК А ГРАФІЧНИЙ МАТЕРІАЛ.....	87
ПЛАКАТ 1. ДІАГРАМА ВАРІАНТІВ ВИКОРИСТАННЯ	88
ПЛАКАТ 2. ER-ДІАГРАМА.....	89
ПЛАКАТ 3. СХЕМИ РОБОТИ АЛГОРИТМІВ.....	90
ПЛАКАТ 4. ДІАГРАМА КЛАСІВ.....	91
ПЛАКАТ 5. ДІАГРАМА ПОСЛІДОВНОСТІ.....	92
ПЛАКАТ 6. ДІАГРАМА КОМПОНЕНТІВ.....	93
ПЛАКАТ 7. ДІАГРАМА РОЗГОРТАННЯ	94

ВСТУП

Сьогодні, одним з найбільш цінних ресурсів людства – є інформація. Вона займає одну з центральних ролей у процесі формування сучасного суспільства. З виникненням Інтернету – основним сховищем інформації стала саме Всесвітня Мережа. Але з часом, її об'єми почали зростати значно швидше, ніж обчислювальні можливості, для обробки даних. Кількість інформації в Інтернеті настільки велика, що людина просто не здатна знайти те, що їй дійсно потрібно.

Сьогодні кількість електронних ресурсів важко підрахувати. Кожен з них представляє собою величезний набір даних. Всю цю інформацію можна і потрібно обробляти. Завдяки цьому можна підвищити ефективність системи. Це відбувається шляхом аналізу поведінки користувачів ресурсів, для прогнозування їх побажань, та відображенню прогнозованих рекомендацій окремим користувачам. Даний тип систем називається «рекомендаційними».

Це інформаційні системи, завдання яких, запропонувати користувачеві дії, послуги або товари на основі попередніх переваг його або користувачів, схожих на нього за купленим товарами / послугами. У зв'язку з поширеністю рекомендаційних систем різних типів, тема є актуальною.

Також актуальною є розробка спеціальної системи, яка буде сама пропонувати користувачу деякі елементи, яка вона буде вважати доцільними. Така система буде формувати свої рекомендації на основі поведінки користувачів в минулому та їх вподобань. Це дозволить користувачам зекономити великий об'єм часу для пошуку необхідного контенту.

Резюмуючи все вищезазначене, можна зробити висновок що тема досліджень в області рекомендаційних систем є досить популярною. Рекомендаційні системи зараз знаходяться у стані стрімкого розвитку, тобто якість їх роботи постійно збільшується. Тому в якості дослідження необхідно провести аналіз найкращих практик, моделей та методів для підвищення ефективності надання рекомендацій.

1 ПРОЕКТНІ РІШЕННЯ З РОЗРОБКИ РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ НА ОСНОВІ ВПОДОБАНЬ КОРИСТУВАЧІВ

1.1 Опис бізнес-процесів

Сьогодні кількість контенту в Всесвітній мережі дійсно вражає. Тому часто нам, звичайним користувачам, досить важко знайти саме те, що нам необхідно. Інколи важко зробити вибір, навіть якщо точно знаєш – що саме ти хочеш. В цей момент нам на допомогу приходять рекомендаційні системи. Можна сказати, що нежива система та алгоритми, на основі яких вона функціонує, знає нас краще, ніж ми самі.

На сьогодні всі найбільші соцмережі, інтернет-магазини та портали мають у своєму арсеналі рекомендаційну систему. В якості рекомендацій може виступати що завгодно (дописи, світлини, відео, товари та ін.). Перші рекомендаційні системи з'явилися ще у 90х роках минулого століття, та свій активний розвиток отримали не так давно. Рекомендаційні систему зробили справжній бум за останні декілька років, вивівши споживання контенту користувачами на істотно новий рівень, зробивши пошук необхідної інформації набагато зручнішим.

1.1.1 Опис процесу діяльності

Кожного дня ми стикаємось з вибором у якомусь сенсі. Рекомендаційні системи створені для того, щоб допомогти користувачам зробити цей вибір, будь-то фільм для вечірнього перегляду чи потенційна друга половинка на порталі для знайомств, адже часто нам не вистачає необхідного досвіду для свідомого вибору.

Для цього система вивчає вподобання інших користувачів для надання індивідуальних рекомендацій. Інколи системи можуть використовувати не лише дані про вподобання інших користувачів, а і неявні критерії, наприклад вік, стать, національність користувача та ін. Саме аналіз таких факторів значно збільшує ефективність роботи рекомендаційної системи.

На відміну від пошукових систем, при роботі з рекомендаційною системою ми не оперуємо з чіткими запитами. В загальному випадку користувачу пропонується

оцінити декілька елементів, для того щоб уникнути так-званої проблеми “холодного старту”, коли нам нічого невідомо про смаки користувача, та на основі цих вподобань, тобто минулого досвіду користувача ми можемо спрогнозувати його вподобання у майбутньому. Це опис роботи рекомендаційної системи в загальному випадку, так як специфікація роботи цілком залежить від конкретного способу її реалізації. Але завжди, при роботі з рекомендаційною системою ми маємо справу з користувачами та множиною альтернатив, на якій користувач повинен зробити свій вибір, при цьому, про існування деяких альтернатив користувач може навіть не здогадуватися.

Таким чином робота рекомендаційної системи являє собою використання певного алгоритму фільтрації існуючих вподобань користувачів, для рекомендування нових наборів альтернатив, що є найбільш релевантними для конкретного користувача.

В деяких випадках функціонування сервісу цілком залежить від рекомендаційної системи. Розглянемо стримінгові музичні сервіси. Наприклад сервіс Яндекс.Музика за грубими підрахунками містить в собі більш ніж 50 років неперервного аудіопотоку. Зрозуміло що ні один користувач не зможе прослухати весь контент, тому пошук нової музики користувачі здійснюють на основі порад друзів, суспільної думки та ін. В задачі рекомендаційної системи на даному етапі входить генерація персоналізованого та релевантного аудіо-потoku для користувача на основі його взаємодії з системою та поведінки в минулому.

Зокрема, у електронній комерції рекомендаційні системи знайшли найбільш широкого застосування. З точки зору бізнесу – головна ціль рекомендаційної системи – збільшення аудиторії користувачів, адже чим більш релевантний контент пропонується користувачам, тим вірогідніше вони залишаться на даному сервісі і тим вірогідніше запропонують даний сервіс іншим. Також, завдяки роботі рекомендаційної системи можуть зрости такі показники ефективності, як: кількість проданих товарів, об’єм продажів, прибуток, середній час перебування користувача на сайті, лояльність користувачів та ін.

Але рекомендаційні системи не завжди використовують лише для рекомендацій товару або контенту користувачам. Наприклад деякі інтернет-магазини використовують їх для збільшення ефективності промо-акцій. Система прогнозує ступінь зацікавленості кожного користувача в деякому товарі, і на основі цього пропонує всім зацікавленим користувачам промо-акцію.

Існує 2 основних типи компаній, що використовують рекомендаційні системи:

- компанія, що зацікавлена у збільшенні об'єму продажів своїх товарів чи послуг. Тобто існують користувачі, які або оцінюють товари, або купляють їх або ні. Тобто необхідно рекомендувати користувачу товар, який буде найбільш релевантним для нього, таким чином збільшивши ймовірність придбання конкретного товару конкретним користувачем. До таких компаній відносяться Netflix, Amazon та ін;
- компанія, що отримує прибуток від реклами. Реклама вбудовується на портал чи інший сайт. При цьому необхідно показувати найбільш релевантну для конкретного користувача рекламу, для підвищення ймовірності переходу користувача за посиланням у рекламі.

Також у рекомендаційної системи існує 2 основних рівні, на яких вона повинна функціонувати:

- глобальні оцінки. Вподобання, що змінюються повільно (музикальні напрями, жанри кіно та літератури та ін), залежність від інформації про користувача (вік, стать, географічні дані) та ін;
- короткочасні тренди. Вподобання що мають дуже великий вплив на істотно малому проміжку часу.

Найпопулярніші рекомендаційні систем:

Amazon – першопроходець та один з лідерів в цій області. Amazon пропонує товари на основі поведінки користувачів(які товари купляли, які оцінювали, які переглядали та ін.)

Netflix – на теренах СНД мало хто користується даним сервісом, адже свою найбільшу популярність він маж саме на Заході, але саме Netflix є компанією, яка на даний момент має найбільш стрімкий розвиток в області надання рекомендацій. Значний вплив на це мала заява компанії про винагороду у розмірі \$1000000 за те, зоб покращити якість роботи алгоритму рекомендацій на 10%. Основа бізнесу Netflix – оренда цифрових копій фільмів.

Last.fm та **Pandora** – музичні рекомендаційні системи. Last.fm для надання рекомендацій використовує комбінацію внутрішніх даних, таких як рейтинги користувачів, з зовнішніми – дані про пісню, дата виходу, ім'я автора, жанр та ін. В свою чергу, Pandora надає рекомендації на основі змісту музичної композиції, використовуючи для цього професійних музикантів, які аналізують композицію по декількох сотнях атрибутів.

Google, Yandex – не дивлячись на те, що дані сервіси являють собою пошукові системи, всі вони використовують функціонал з надання рекомендацій. Вони намагаються спрогнозувати, наскільки даний сайт релевантний до даного пошукового запиту. Також майже у всіх масштабних пошукових систем є багато побічних проєктів, що являюь собою рекомендаційні системи (Google.Music, Yandex.Music і т.д.)

1.1.2 Актори і функції

Наведемо список і функцій системи у вигляді діаграми варіантів використання на рисунку 1.1 та у додатку А:

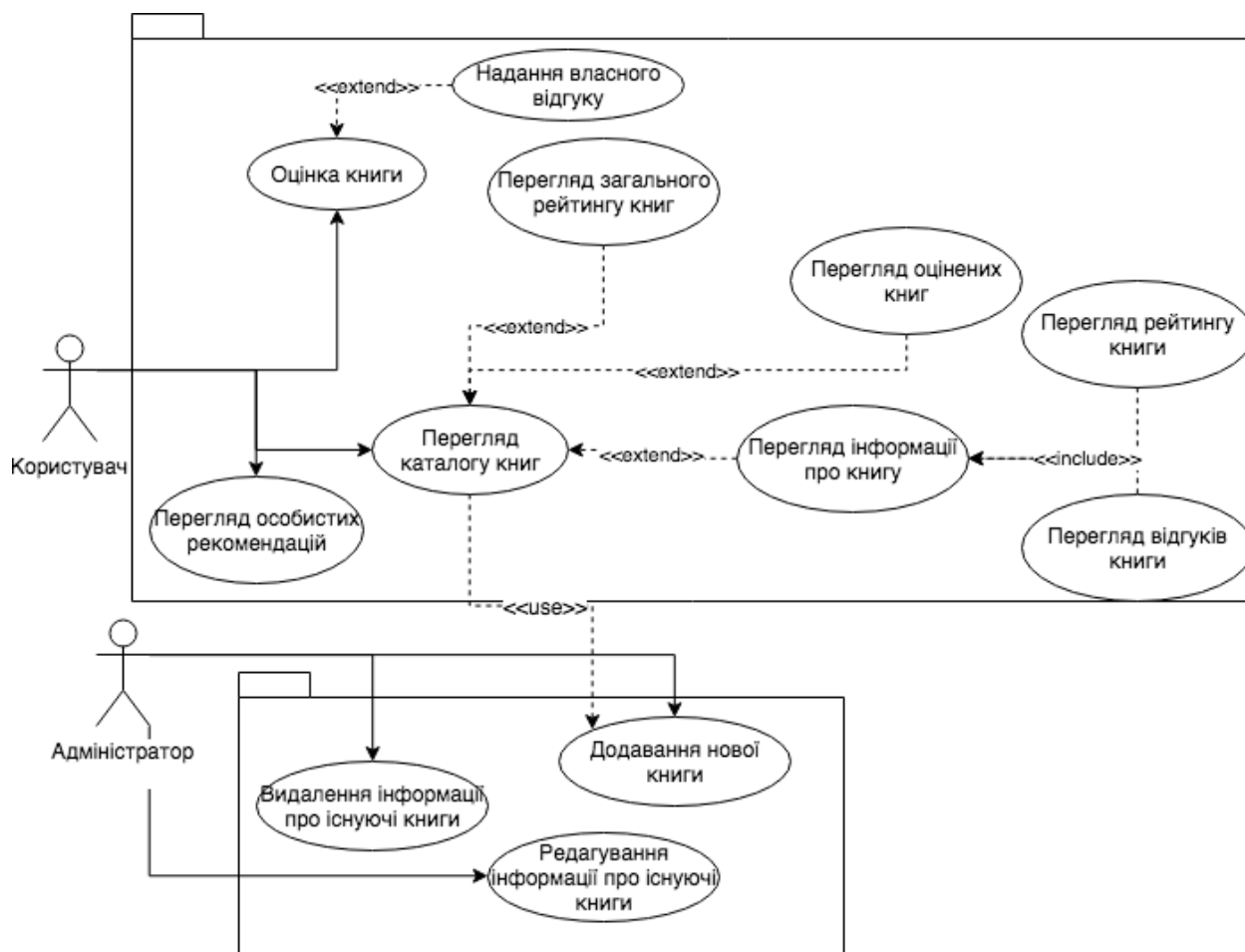


Рисунок 1.1 – Діаграма варіантів використання рекомендаційної системи

Як видно з діаграми – в системі існують 2 актори – **користувач** та **адміністратор**. Кожен з акторів може взаємодіяти з системою наступним чином:

Адміністратор:

- керування інформацією про існуючі книги.

Користувач:

- перегляд каталогу книг;
- перегляд особистих рекомендацій;
- оцінка книги.

Опишемо вищезазначені варіанти використання більш детально.

Варіанти використання системи адміністратором описані у таблиці 1.1:

Таблиця 1.1 – Варіанти використання для актора **Адміністратор**

Актор	Назва пакету варіантів	Варіанти використання
Адмін істрат ор	Керування інформацією про існуючі книги	Додавання інформації про нову книгу
		Редагування інформації про книгу
		Видалення інформації про книгу

Варіанти використання системи користувачем описані у таблиці 1.2:

Таблиця 1.2 – Варіанти використання для актора **Користувач**

Актор	Головний варіант	Підлеглі варіанти
Користувач	Перегляд каталогу книг	Перегляд загального рейтингу книг
		Перегляд оцінених книг
		Перегляд інформації про книгу
		Перегляд відгуків на книгу
	Оцінка книги	Надання власного відгуку
	Перегляд особистих рекомендацій	

Опишемо кожний з варіантів використання більш детально:

Керування інформацією про існуючі книги – адміністратор займається веденням реєстру книг у системі. Для цього він використовує спеціальний

програмний інструмент – панель адміністратора. Це надає йому можливість управління інформацією про книги, що наявні у системі, а саме - заведення інформації про нову книгу, редагування, перегляд або видалення інформації про існуючі книги.

Перегляд каталогу книг – користувач має можливість переглянути загальний каталог книг, не використовуючи рекомендації системи. Тобто каталог представляє собою загальний список, який включає в себе всі існуючі елементи(книги) в системі. Також користувач має можливість переглянути детальну інформацію для кожної книги в каталозі, а також ознайомитися з відгуками. Крім того перегляд каталогу включає в себе можливість фільтрації елементів, таким чином користувач має можливість ознайомитися зі списком оцінених книг.

Оцінка книги – система надає можливість надання оцінки для кожної книги окремим користувачем. Тобто користувач має можливість надати кожному елементу кількісну (оцінка за числовою шкалою), чи якісну (оцінка за загальним враженням) характеристику. Дані оцінки впливають як на загальний рейтинг кожного елемента в системі, так і на вірогідність рекомендації даного елемента деяким користувачам системи.

Перегляд особистих рекомендацій – користувач має можливість ознайомитися зі списком рекомендованих елементів, які система вважає найбільш релевантними.

1.1.3 Структура бізнес-процесів

Основні процеси діяльності системи включають в себе – ведення реєстру книг, оцінка книг користувачами та надання рекомендацій.

Загальний опис усіх бізнес-процесів системи наведено на діаграмі діяльності і міститься у графічних матеріалах.

Для початку роботи системи адміністратору необхідно наповнити систему контентом, тобто завести реєстр книг. Після того, як система міститиме в собі достатньо контенту, користувачі матимуть змогу переглядати загальний каталог.

Перед початком роботи з системою користувачі мають змогу пройти процес авторизації. Але система також включає в себе варіант взаємодії користувача у якості Гостя. При цьому доступ до ключових можливостей системи, такі як оцінка елементів, написання власного відгуку а також перегляд особистих рекомендацій буде відсутній. У цьому випадку неавторизований користувач матиме доступ лише до базового функціоналу системи, а саме – перегляд каталогу, перегляд детальної інформації про книги та перегляд загального рейтингу. Тому більш детально розглянемо лише випадок успішної авторизації користувача на сервісі.

Користувач обирає елемент, що його зацікавив, таким чином потрапляючи на сторінку з детальною інформацією. Детальна інформація містить в собі все необхідні додаткові дані, які можуть допомогти користувачу у формуванні першого враження від книги (жанр, автор, рік випуску, кількість сторінок і т.д.).

Користувач має можливість ознайомитися з рейтингом книги, який заснований на оцінках інших користувачів, а також переглянути відгуки для даного елемента. Також користувач має можливість поставити власну оцінку для кожного елемента та залишити власний відгук.

Окремо від перегляду загального каталогу, система надає користувачу рекомендації, щодо елементів, які вважає найбільш релевантними для нього. Дані рекомендації засновані на оцінках інших користувачів, які за власними вподобаннями є найбільш наближеними до користувача, якому необхідно надати рекомендації.

Також, в якості параметрів, що впливають на рекомендації, можуть виступати як дані про користувачів (вік, стать і т.д.), так і дані про елементи безпосередньо (жанр, кількість сторінок і т.д.).

У графічному матеріалі наведено більш детальний опис роботи системи.

1.2 Опис постановки задачі

Система призначена для надання рекомендацій користувачам щодо найбільш релевантних для них елементів.

Ціль створення: Максимально повне задоволення потреб користувачів у підборі релевантних для них елементів шляхом урахування рекомендацій інших користувачів, або рекомендацій щодо інших елементів.

Задачі системи. Для реалізації поставлених цілей система повинна вирішувати такі задачі:

- а) надання можливостей ведення реєстру книг:
 - 1) заведення нової інформації про книгу;
 - 2) редагування існуючої інформації;
 - 3) видалення існуючої інформації;
- б) надання можливостей ознайомлення з контентом системи
 - 1) перегляд каталогу елементів;
 - 2) підрахунок загального рейтингу елементів у системі;
- в) надання можливостей оцінки елементів системи:
 - 1) оцінка елементів системи за прийнятою шкалою;
 - 2) написання відгуку на конкретний елемент;
- г) надання рекомендацій конкретному користувачу:
 - 1) вибір найбільш релевантних елементів системи для кожного конкретного користувача.

1.3 Огляд аналогів системи, що розробляється

На сьогодні існує безліч рекомендаційних систем. Їх можна певним чином згрупувати за контентом, щодо якого надаються рекомендації. А саме:

- аудіо контент;
- відео контент;
- товари;
- послуги;
- новини і т.д.

Так як рекомендаційна система фокусується лише на конкретному виді контенту, всі вони певним чином відрізняються одна від одної.

Розглянемо найпопулярніших представників для кожного типу контенту:

Spotify – найпопулярніший сервіс для відтворення цифрового аудіо-потoku. Налічує в собі найбільшу аудіо-бібліотеку, що дозволяє легально і безкоштовно прослухати музичні композиції. Spotify є найпершим стримінговим сервісом, що дозволяє слухати музику онлайн. Сервіс доступний в Америці, в більшості країн Європи, деяких країнах Азії, а також в Австралії та Нової Зеландії. Має потужну рекомендаційну модель;

YouTube - відеохостинговий сайт, що надає користувачам послуги зберігання, доставки та показу відео. Користувачі можуть завантажувати, переглядати, оцінювати, коментувати, додавати до вибраного та ділитися темою або іншими відеозаписами. Завдяки простоті та зручності використання YouTube став найпопулярнішим відеохостингом і другим сайтом у світі за кількістю відвідувачів. Надає своїм користувачам рекомендації 2х типів: 1) світові тренди, тобто відео, що є найпопулярнішими в даний момент; 2) особисті рекомендації – відео які пропонуються на основі минулих переглядів конкретного користувача;

Amazon - американська компанія, найбільша у світі в торгівлі товарами та послугами через Інтернет та є одним з перших інтернет-сервісів, орієнтованих на продаж реальних товарів масового попиту. Штаб-квартира розташована в Сіетлі

(штат Вашингтон). Amazon рекомендує товари, на основі того, що ви купили, що переглянули, які рейтинги ставили, які залишили відгуки;

Google News - безкоштовний агрегатор новин, який надається та управляється компанією Google Inc. Сервіс створено в 2002 році. Рекомендації надаються на основі даних користувача (вік, стать, національність) та на основі минулих пошукових запитів у системі Google.

Щодо відомих рекомендаційних систем, які направлені на книги – найбільш відомим був система Імхонет . Це інтелектуальне середовище розваг, експертну роботу в якій виконують самі користувачі. Проект був заснований у 2007 році і його першим розділом був саме розділ “Книги”. Але у квітні 2017 року сервіс припинив своє існування через невиправдання очікувань інвесторів. Таким чином користувачі сервісу втратили свої оцінки та відгуки за всі 10 років роботи системи.

Виходячи з вищезазначеного можна зрозуміти, що зараз ніша рекомендаційної системи книг є вільною (хоча б на теренах СНД). Саме тому розробка нового сервісу є дуже доцільною, особливо якщо врахувати розвиток рекомендаційних систем за останні декілька років.

1.4 Рішення з інформаційного забезпечення

При розробці даної системи буде використовуватись реляційна база даних. Тому зобразимо структуру бази даних на концептуальному рівні у вигляді ER-діаграми, зображеної на рисунку 1.2 та у додатку А:

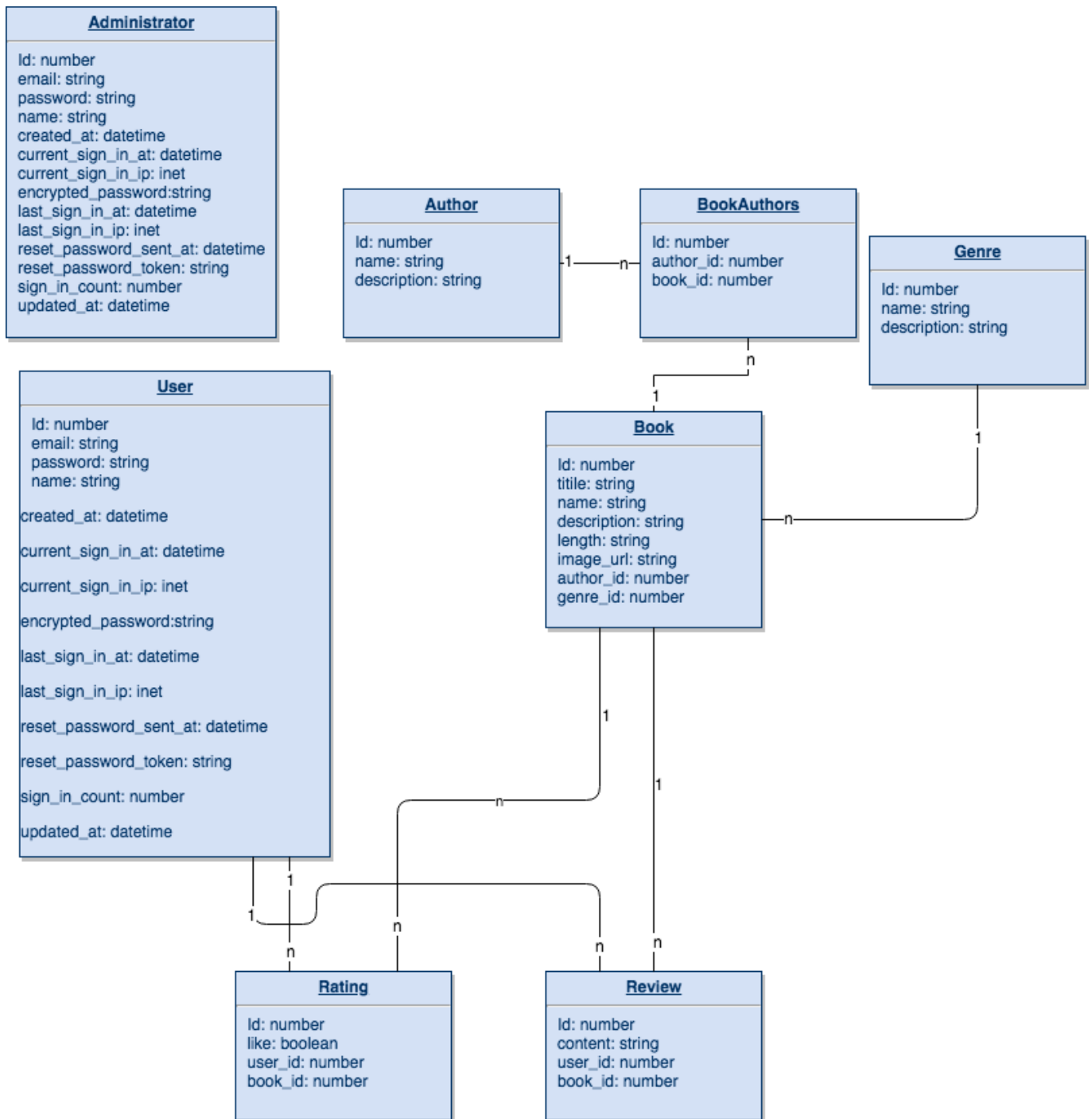


Рисунок 1.2 – ER-діаграма бази даних

Наведемо опис таблиць, що використовуються для зберігання даних всередині системи та відображені на вищезазначеній діаграмі у вигляді таблиці 1.3.

Таблиця 1.3 – Таблиці бази даних

Назва таблиці	Опис
Administrator	Інформація про адміністратора системи
User	Інформація про користувача
Book	Інформація про книгу
Author	Інформація про автора книги
Genre	Інформація про жанр книги
Rating	Інформація про оцінку книги користувачем
Review	Інформація про відгук користувача на книгу

Нижче наведено детальний опис таблиць що використовуються.

Опис полів таблиці Administrator наведено у таблиці 1.4:

Таблиця 1.4 – Таблиця Administrator

Параметр	Тип	Опис	Обмеження
created_at	datetime	Дата створення запису	Не може бути порожнім
current_sign_in_at	datetime	Дата поточного входу в систему	

Продовження таблиці 1.4

Параметр	Тип	Опис	Обмеження
current_sign_in_ip	inet	IP-адреса поточного входу в систему	
email	string	Email адміністратора	Не може бути порожнім
encrypted_password	string	Пароль адміністратора	Не може бути порожнім
id	integer	ID адміністратора	Не може бути порожнім. Первинний ключ
last_sign_in_at	datetime	Дата останнього входу в систему	
last_sign_in_ip	inet	IP-адреса останнього входу в систему	
reset_password_sent_at	datetime	Дата останнього відновлення паролю	
reset_password_token	string	Ключ для відновлення паролю	

Параметр	Тип	Опис	Обмеження
sign_in_count	integer	Кількість входів у систему	
updated_at	datetime	Дата останнього оновлення запису	

Опис полів таблиці User наведено у таблиці 1.5:

Таблиця 1.5 – Таблиця User

Параметр	Тип	Опис	Обмеження
created_at	datetime	Дата створення запису	Не може бути порожнім
current_sign_in_at	datetime	Дата поточного входу в систему	
current_sign_in_ip	inet	IP-адреса поточного входу в систему	
email	string	Email Користувача	Не може бути порожнім

Параметр	Тип	Опис	Обмеження
encrypted_password	string	Пароль Користувача	Не може бути порожнім
id	integer	ID Користувача	Не може бути порожнім. Первинний ключ
last_sign_in_at	datetime	Дата останнього входу в систему	
last_sign_in_ip	inet	IP-адреса останнього входу в систему	
reset_password_sent_at	datetime	Дата останнього відновлення паролю	
reset_password_token	string	Ключ для відновлення паролю	
sign_in_count	integer	Кількість входів у систему	

Продовження таблиці 1.5

Параметр	Тип	Опис	Обмеження
updated_at	datetime	Дата останнього оновлення запису	
name	string	Ім'я користувача	Не може бути порожнім.

Опис полів таблиці Author наведено у таблиці 1.6:

Таблиця 1.6 – Таблиця Author

Параметр	Тип	Опис	Обмеження
id	number	ID автора	Не може бути порожнім. Первинний ключ
name	string	Ім'я користувача	Не може бути порожнім
description	string	Додаткова інформація про автора	

Опис полів таблиці Genre наведено у таблиці 1.7:

Таблиця 1.7 – Таблиця Genre

Параметр	Тип	Опис	Обмеження
id	number	ID автора	Не може бути порожнім. Первинний ключ
name	string	Назва жанру	Не може бути порожнім
description	string	Додатковий опис жанру	

Опис полів таблиці Book наведено у таблиці 1.8:

Таблиця 1.8 – Таблиця Book

Параметр	Тип	Опис	Обмеження
id	number	ID книги	Не може бути порожнім. Первинний ключ
title	string	Назва книги	Не може бути порожнім
description	string	Детальний опис книги	

Параметр	Тип	Опис	Обмеження
year	string	Рік написання книги	
length	number	Кількість сторінок	
image_url	string	Посилання на зображення обкладинки	
author_id	number	Ідентифікатор автора книги	
genre_id	number	Ідентифікатор жанру книги	Не може бути порожнім. Зовнішній ключ
arrival_time	datetime	Час прибуття	Не може бути порожнім. Зовнішній ключ

Опис полів таблиці Book наведено у таблиці 1.9:

Таблиця 1.9 – Таблиця Review

Параметр	Тип	Опис	Обмеження
id	number	ID відгуку	Не може бути порожнім. Первинний ключ

Продовження таблиці 1.9

Параметр	Тип	Опис	Обмеження
content	string	Текст відгуку	Не може бути порожнім.
user_id	number	Ідентифікатор автора відгуку	Не може бути порожнім. Зовнішній ключ
book_id	number	Ідентифікатор книги на яку написаний відгук	Не може бути порожнім. Зовнішній ключ

Опис полів таблиці Rating наведено у таблиці 1.10:

Таблиця 1.10 – Таблиця Rating

Параметр	Тип	Опис	Обмеження
id	number	ID оцінки	Не може бути порожнім. Первинний ключ
like	boolean	Ідентифікатор оцінки	Не може бути порожнім.
user_id	number	Ідентифікатор автора оцінки	Не може бути порожнім. Зовнішній ключ

Параметр	Тип	Опис	Обмеження
book_id	number	Ідентифікатор книги на яку надано оцінку	Не може бути порожнім. Зовнішній ключ

Опис полів таблиці BookAuthors наведено у таблиці 1.11:

Таблиця 1.11 – Таблиця BookAuthors

Параметр	Тип	Опис	Обмеження
id	number	ID оцінки	Не може бути порожнім. Первинний ключ
book_id	number	Ідентифікатор книги	Не може бути порожнім. Зовнішній ключ
author_id	number	Ідентифікатор автора оцінки	Не може бути порожнім. Зовнішній ключ

Висновок до розділу

В результаті виконання даного розділу був здійснений опис основних бізнес-процесів та процесів діяльності. Виділені актори, які будуть взаємодіяти з системою, а також описані варіанти використання системи акторами. Також був здійснений опис постановки задачі та огляд основних аналогів системи, що розробляється. У подальшому це дозволить використати сильні сторони аналогів при розробці власної системи. Також були здійснені рішення з інформаційного забезпечення у вигляді ER-діаграми бази даних та детального опису кожної з таблиць.

2 МОДЕЛІ ТА МЕТОДИ ФОРМУВАННЯ РЕКОМЕНДАЦІЙ

2.1 Змістовна постановка задачі

Рекомендаційні системи, для реалізації коректної роботи застосовують комплексний підхід, який включає в себе застосування принципів та напрацювань з декількох галузей знань. Зокрема: когнітивістика, прогнозування, машинне навчання, інформаційний пошук і т.д. Також значного впливу рекомендаційні системи зазнали з боку бізнесу, адже результати маркетингових досліджень є не менш цінними, ніж інформаційний аналіз даних. Але завжди – рекомендаційні системи використовувалися для знаходження рейтингів об'єктів для конкретного користувача. Це може використовуватися як суто нова інформація, тобто передбачення вподобань користувача щодо нових елементів, так і для коригування роботи системи, за рахунок порівняння реальної оцінки користувача з прогнозованою.

Для роботи рекомендаційної системи необхідні дані, на яких будуть базуватися рекомендації. Перш за все це дані про користувачів та елементів всередині системи. Повний цикл роботи рекомендаційної систему повинен брати початок на етапі отримання інформації і закінчувати наданнями рекомендацій користувачам. Від інформації, якою буде оперувати система залежить вибір алгоритму, завдяки яким будуть будуватися прогнози.

Таким чином, рекомендаційна система – це система, націлена на передбачення вподобань користувачів щодо елементів, що виступають у якості основного контенту системи (фільми, книги, музичні твори, новини та ін.), за рахунок відомої інформації користувача або елементів всередині системи [1].

Формально – змістовну постановку задачі можна представити у вигляді IDEF0 діаграми (рисунок 2.1):



Рисунок 2.1 – IDEF0 діаграма роботи системи

2.2 Математична модель

Математично, задачу знаходження найбільш релевантного об'єкта можна описати наступним чином [2,3]:

$$\forall u \in U, s'_u = \operatorname{argmax}_{s \in S} h(u, s), \quad (2.1)$$

де U – множина користувачів, u – вибраний користувач, S – множина елементів системи, які потенційно можуть бути рекомендовані користувачу, s – вибраний елемент, h – функція, яка показує ступінь зіставлення деякого елементу s з деяким користувачем u .

Таким чином, задача зводиться до знаходження такого елементу $s' \in S$, для якого ступінь зіставлення з користувачем $u \in U$ є максимальним (у випадку рекомендації одного елементу).

При цьому на точність рекомендації суттєво впливає не лише інформація всередині системи, а і вибір функції h .

2.3 Огляд методів розв’язання

Для розв’язання вищезазначеної задачі існує декілька підходів. Зокрема основні з них [3]:

- фільтрація на основі вмісту (item-based);
- колаборативна фільтрація (user-based);
- гібридний підхід.

Розглянемо кожен з підходів більш детально.

2.3.1 Фільтрація на основі вмісту

Як можна зрозуміти з назви підходу, фільтрація на основі вмісту перш за все опирається на інформацію про контент системи, а не про користувачів. Тобто рекомендації базуються на знаходженні схожих елементів, до тих, що користувач вже оцінив в минулому. Для цього необхідно створити профіль користувача та профіль елемента. Після цього, на основі параметрів елементів системи, можна зробити висновок щодо відповідності конкретного елемента конкретному користувачу. Для опису елементів системи та створення їх профілю рекомендаційні системи ставлять у відповідність кожному елементу певний набір ключових слів. Формально роботу рекомендаційної системи на основі фільтрації вмісту можна зобразити наступним чином (рисунок 2.2 та додаток А):

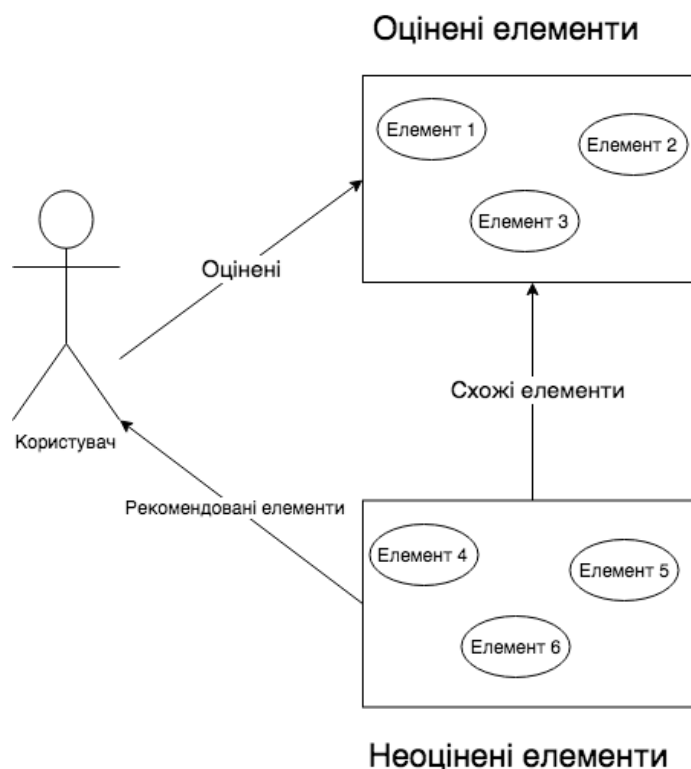


Рисунок 2.2 – Принцип роботи рекомендаційної системи на основі фільтрації вмісту

У рекомендаційних системах з фільтрацією на основі вмісту функція зіставлення $h(u, s)$ деякого користувача u з деяким елементом s використовує інформацію про інші елементи системи $s' \in S$, які користувач u вже оцінив в минулому та схожість цих елементів з даним елементом s [3].

Тобто, в загальному випадку, для того щоб рекомендувати користувачу нові елементи, система повинна проаналізувати елементи, які користувач високо оцінив в минулому, знайти ознаки, що об'єднують ці елементи, та на основі знайдених ознак знайти нові елементи, які їм відповідають, і які не мають оцінок від користувача, тобто є новими для нього.

Формально – якщо елемент A сподобався користувачу U , а елемент B схожий на елемент A , то можна зробити висновок, що елемент B також сподобається користувачу U .

Таким чином, користувацький профіль формується у вигляді параметрів, що характеризують кожен елемент $s' \in S'$, де S' – множина оцінених елементів. У

якості таких параметрів найчастіше виступають ключові слова, також вагові коефіцієнти ключових слів для кожного з елементів системи. Одним з найчастіше використовуваних способів знаходження даних вагових коефіцієнтів є TD-IDF міра [4].

Її можна обчислити за наступною формулою:

$$TF_{ij} = \frac{f_{ij}}{\max_z f_{zj}}, \quad (2.2)$$

де f_{ij} - кількість входжень деякого ключового слова k_i в елемент d_j , при цьому ключове слово k_j зустрічається в n_i об'єктах.

Але при використанні вищезазначеної формули враховується лише частоти входження ключового слова. Це може призвести до випадку, коли максимальну вагу будуть мати найбільш розповсюджені ключові слова, що в подальшому може призвести до некоректного прогнозування вподобань конкретного користувача. Для уникнення такого випадку використовується величина IDF_i , яка є зворотною до частоти входження ключового слова в елемент.

$$IDF_i = \log \frac{N}{n_i}, \quad (2.3)$$

де N – кількість елементів, які потенційно можуть бути рекомендовані користувачу.

Виходячи з вищезазначених формул, можемо обчислити вагу w_{ij} ключового слова k_i в елементі d_j наступним чином:

$$w_{ij} = TF_{ij} * IDF_i \quad (2.4)$$

Тоді профіль елементу d_j можна задати наступним чином:

$$Content(d_{ij}) = (w_{1j}, \dots, w_{kj}) \quad (2.5)$$

Рекомендаційні системи на основі фільтрації контенту при формуванні рекомендацій використовують інформацію про елементи, що вже були високо оцінені користувачем в минулому. З множини всіх елементів систему обираються лише ті, які є найбільш подібними до елементів, що вже були оцінені, таким чином будується гіпотеза, що нові елементи будуть також високо оцінені користувачем. Даний набір елементів, що вже були оцінені користувачем формують його профіль, а також вектор ваг ключових слів:

$$(w_{ul}, \dots, w_{uk}), \quad (2.6)$$

де кожна вага w_{ui} визначає важливість ключового слова k_i для користувача u .

Тоді профіль елемента та профіль користувача можна представити у вигляді TF-IDF векторів \vec{w}_s та \vec{w}_u , при цьому функція зіставлення елемента та користувача $h(u, s)$ може бути представлена у вигляді косинусу кута між векторами \vec{w}_s та \vec{w}_u [5]:

$$h(u, s) = \cos(\vec{w}_u, \vec{w}_s) = \frac{\vec{w}_u \vec{w}_s}{\|\vec{w}_u\| \|\vec{w}_s\|} = \frac{\sum_{i=1}^K w_{iu} w_{is}}{\sqrt{\sum_{i=1}^K w_{iu}^2} \sqrt{\sum_{i=1}^K w_{is}^2}}, \quad (2.7)$$

де K – загальна кількість ключових слів в системі.

Рекомендаційні системи часто базуються не лише на використанні деякої евристики та методів інформаційного пошуку, а також на використанні інших технік, таких як наївний байєсовський класифікатор, нейронні мережі, дерева рішень, кластеризація та інші техніки машинного навчання. Наприклад, рекомендаційна система, використовуючи оцінки елементів користувачем, за допомогою наївного баєсовського класифікатора може визначити елементи, які користувач ще не оцінив, та обчислити ймовірність належності певного елемента s_i до певного класу C_i (високо оцінені/низько оцінені елементи).

2.3.2 Колаборативна фільтрація

Рекомендаційні системи, в основі роботи яких лежить колаборативна фільтрація на відміну від фільтрації на основі вмісту для своєї роботи аналізують дані про користувачів, а не про елементи системи. Кожному користувачу у відповідність ставиться деяка група користувачів, зі схожими з ним смаками. На основі цього формулюється гіпотеза, що користувачі, які однаково оцінили деякі об'єкти в минулому, швидше за все, однаково оцінять інші об'єкти в майбутньому [3]. Формально роботу рекомендаційної системи на основі колаборативної фільтрації можна зобразити наступним чином (рисунок 2.3 та додаток А):

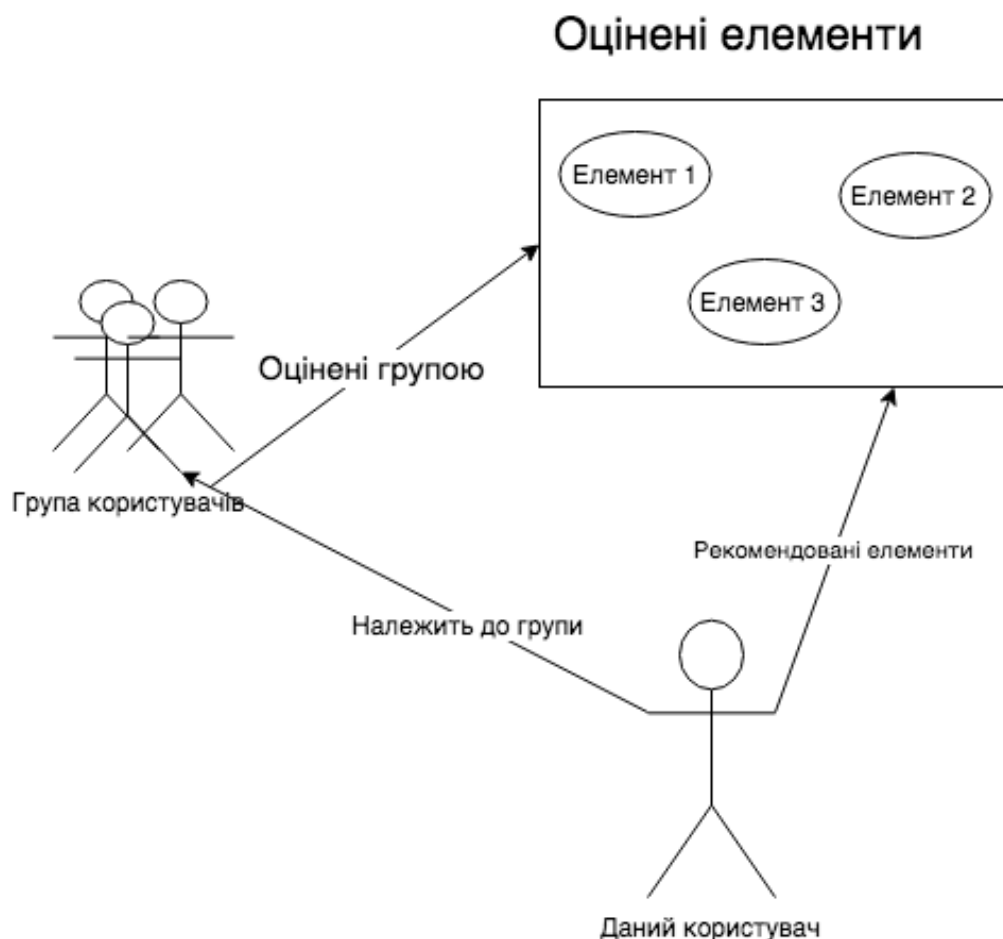


Рисунок 2.3 – Принцип роботи рекомендаційної системи на основі колаборативної фільтрації

Тобто, методи колаборативної фільтрації для побудови рекомендацій використовують інформацію про оцінки інших користувачів, які належать до одної групи. В цьому випадку функція $h(u, s)$ задає ступінь зіставлення елементу s з користувачем u , та обчислюється в залежності від зіставлення $h(u_j, s)$ для елементу s для користувачів $u_j \in U'$, де U' - деяка група користувачів [6].

Для визначення ступеня схожості користувачів та формування груп, необхідно побудувати профіль користувача для кожного з них. При побудові профіля може використовуватися як дані про вподобання користувачів, так і більш загальна інформація, наприклад вік, стать, національність, профілі в соціальних мережах та ін. Подібна загальна інформація про користувача є дуже важливою на перших етапах роботи користувача з рекомендаційною системою, коли користувач ще не встиг оцінити достатньо елементів, для формування вподобань його профіля. Тобто маємо випадок, коли система нічого не знає про користувача, і не може побудувати для нього коректні рекомендації. Подібна проблема має назву “холодний старт”. Для зменшення впливу даної проблеми і необхідно використовувати загальні дані про користувача, адже на основі них також можна віднести його до деякої групи.

На сьогодні існує безліч рекомендаційних систем, які працюють на основі колаборативної фільтрації. Однією з перших була система “Grundy system” [7]. Дана система являла собою програму-бібліотекаря, яка в процесі формування рекомендацій, пропонувала користувачам відповісти на низку запитань для формування профілю користувача. Також при наданні елементу оцінки користувач повинен був відповісти на уточнюючі запитання, для більшої ґрунтовності свого вибору. Тобто, якщо деякий користувач негативно оцінив елемент, система намагається дізнатися причини такої низької оцінки, щоб в майбутньому уникнути рекомендації подібних елементів. Тим часом система “Tapestry» [8] пропонувала користувачам власноруч знайти інших користувачів, вподобання яких, на їх думку, збігалися.

Методи колаборативної фільтрації можна поділити на 2 типи:

- методи, засновані на пам'яті;
- методи, засновані на моделях.

Методи, засновані на пам'яті, використовують дані про всі елементи, які користувач оцінив в минулому, для передбачення рейтингу нового елементу [9]:

$$r_{us} = aggr_{u \in U'} r_{u's} , \quad (2.8)$$

де r_{us} - передбачений рейтинг елементу s для користувача u , U' - множина користувачів, які оцінили елемент s , та входять в одну групу з користувачем u .

В якості функції агрегування можна використовувати різні підходи. Найпростіший з них – загальне середнє:

$$r_{us} = \frac{1}{N} \sum_{u \in U'} r_{u's} , \quad (2.9)$$

де N – потужність множини U' .

Частіше для агрегування застосовуються наступний вираз:

$$r_{us} = k \sum_{u \in U'} sim(u, u') * r_{u's} , \quad (2.10)$$

де k – параметр для нормалізації, який обчислюється наступним чином:

$$k = \frac{1}{\sum_{u \in U'} sim|u, u'|} \quad (2.11)$$

Тут функція $sim(u, u')$ - міра схожості (англ. "similarity"). Дана величина є зворотною до відстані і в більшості випадків використовується в якості вагових коефіцієнтів. Тобто, в залежності від схожості користувачів u, u' , ступінь впливу оцінок користувача u' на рекомендації для користувача u буде варіюватися. У випадку використання зважених сум (2.10) неможливо враховувати деякі деталі. Наприклад – не всі користувачі оцінюють елементи з однаковим рівнем

відповідальності. Для уникнення даної проблеми використовується наступний підхід [9]:

$$r_{us} = \bar{r}_u + k \sum_{u \in u'} \text{sim}(u, u') * (r_{u's} - \bar{r}_{u'}) \quad (2.12)$$

$$\bar{r}_u = \frac{1}{S_u} \sum_{s \in S_u} r_{us} \quad (2.13)$$

Даний вираз включає в себе відмінності від загального рейтингу для конкретного користувача, замість використання абсолютних оцінок.

Для обчислення міри схожості $\text{sim}(u, u')$ користувачів рекомендаційні системи використовують різноманітні підходи. В більшості систем схожість користувачів перш за все визначається схожістю оцінок, які вони дали окремим елементам. Власне в якості міри, найбільш використовувані 2 методи:

- знаходження кореляції;
- знаходження косинусних коефіцієнтів.

Нехай $S_{uu'}$ - множина всіх елементів, які оцінили користувачі u та u' . Тоді $S_{uu'}$ можна задати наступним чином [10]:

$$S_{uu'} = \{s \in S \mid r_{u's} \neq 0, r_{us} \neq 0\} \quad (2.14)$$

При використанні колаборативної фільтрації $S_{uu'}$ використовується у якості проміжного параметру, для виявлення найбільш схожих користувачів, та може обчислюватися багатьма методами, найпростіший з яких – перетин множин $S_{u'}$ та S_u . Деякі методи [11] дозволяють знайти найближчих сусідів користувача u без обчислення $S_{uu'}$. Для цього використовується коефіцієнт Пірсона [12,13]:

$$\text{sim}(u, u') = \frac{\sum_{s \in S_{uu'}} (r_{us} - \bar{r}_u)(r_{u's} - \bar{r}_{u'})}{\sqrt{\sum_{s \in S_{uu'}} (r_{us} - \bar{r}_u)^2 \sum_{s \in S_{uu'}} (r_{u's} - \bar{r}_{u'})^2}} \quad (2.15)$$

В методах, які використовують косинусний коефіцієнт [14,15] користувачі u, u' представлені у вигляді векторів в m -мірному просторі, де $m = |S_{uu'}|$. Таким чином міра схожості може бути обчислена за допомогою наступної формули:

$$\text{sim}(u, u') = \cos(\vec{u}, \vec{u'}) = \frac{\vec{u} \cdot \vec{u'}}{\|\vec{u}\| \cdot \|\vec{u'}\|} = \frac{\sum_{s \in S_{uu'}} r_{us} r_{u's}}{\sqrt{\sum_{s \in S_{uu'}} r_{us}^2} \sqrt{\sum_{s \in S_{uu'}} r_{u's}^2}} \quad (2.16)$$

Можна помітити, що косинусний коефіцієнт використовується як в рекомендаційних системах, що використовують фільтрацію на основі вмісту, так і в тих, що застосовують колаборативну фільтрацію.

Існує безліч модифікацій та вдосконалень які значно збільшують ефективність роботи даних методів [16,17]. Наприклад зумовлені рейтинги [18] є одним з модифікацій методу, що заснований на пам'яті.

Але вищезазначені підходи також використовувались для обчислення міри схожості користувачів. В. Sarwar з співавторами [19] використали ті ж самі підходи для знаходженні кореляції між елементами, та на основі цього обчислили передбачені оцінки для них. Також існує емпіричне доведення, що алгоритми, які засновані на використанні фільтрації змісту здатні працювати швидше та використовувати меншу кількість ресурсів, при цьому забезпечуючи не меншу точність рекомендацій, порівняно з алгоритмами, що використовують колаборативну фільтрацію.

2.3.3 Гібридний підхід

Деякі рекомендаційні системи застосовують гібридний підхід для формування рекомендацій. Гібридність полягає у поєднанні результатів двох різних підходів – фільтрації на основі вмісту та колаборативної фільтрації. Завдяки цьому можна, в деякій мірі, уникнути недоліків вищезазначених підходів. Існує декілька основних варіантів комбінування підходів до фільтрації [3]:

- включення характеристик елементів в колаборативну фільтрацію;
- включення характеристик користувачів в фільтрацію за вмістом;

- побудова єдиної моделі на основі колаборативної фільтрації та фільтрації за вмістом.

2.3.4.1 Включення характеристик елементів в колаборативну фільтрацію

Такі системи працюють, перш за все, використовуючи колаборативну фільтрацію, але включають у профіль користувача деякі дані про елементи. Дані про властивості елементів використовуються при знаходженні міри схожості користувачів. Переваги даного підходу полягають у тому, що він дозволяє підвищити релевантність рекомендацій, коли у сервісі недостатньо пар користувачів з достатньо великою кількістю спільних оцінених елементів. Також цей метод дозволяє рекомендувати користувачу не лише елементи, які сподобались схожим на нього іншим користувачам, а також елементи, які схожі на ті, що даний користувач оцінив раніше.

2.3.4.2 Включення характеристик користувачів в фільтрацію за вмістом

Найбільш відомий метод для реалізації даного підходу – зниження розмірності профілей, що використовують фільтрацію за вмістом. В одному з таких методів [20] використовується латентно-семантичний аналіз LSA для створення колаборативного змісту профілей користувачів. Даний підхід є значно ефективнішим, так як рекомендації будуються не лише за рахунок схожості елементів.

2.3.4.3 Побудова єдиної моделі на основі колаборативної фільтрації та фільтрації за вмістом

За останні роки даний підхід здобув значну популярність [19,20]. Суть методу полягає у тому, що в одній системі використовуються характеристика колаборативної фільтрації та фільтрації за вмістом (наприклад кількість сторінок книги та вік користувача). Одним з методів для реалізації даного підходу є уніфікований імовірнісний метод [21], який використовує латентно-семантичний аналіз [22].

Гібридні методи фільтрації можна модифікувати та доповнювати іншими методами для підвищення якості рекомендацій.

Існують роботи [18, 23, 24, 25], які свідчать про те, що релевантність рекомендацій, що побудовані за допомогою гібридних методів фільтрації, є значно вищою за рекомендацій, що побудовані лише за допомогою колаборативної фільтрації чи фільтрації за вмістом.

2.4 Модифікація методу розв'язання задачі

Модифікуємо алгоритм надання рекомендацій. Для цього розділимо метод колаборативної фільтрації на 2 окремих:

- фільтрація на основі сусідства;
- демографічна фільтрація.

В методі фільтрації на основі сусідства, рейтинг елементу z для користувача u прогнозується за рахунок рейтингів користувачів, які, на думку системи, мають схожі до даного користувача вподобання. Даний підхід залежить від обраної кількості сусідів користувача u , оцінки яких будуть враховуватися при прогнозуванні рейтингу. Власне ступінь сусідства обчислюється за допомогою міри схожості (косинусна міра, кореляція).

Демографічні характеристики користувачів відіграють дуже важливу роль при формуванні груп користувачів для надання рекомендацій. Рекомендаційні системи часто зіштовхуються з згаданою вище проблемою “холодного старту”, через недостатню кількість даних про оцінки користувачів [26]. В таких випадках необхідно використовувати дані, які доступні з самого початку роботи користувача з системою. Демографічні дані про користувача відносяться до такого типу даних, адже з великою вірогідністю користувач заповнить свій профіль на самому початку роботи з системою. Завдяки цьому система зможе формувати рекомендації більш точно, адже, знаючи вік, стать чи національність користувача значно легше віднести його до певної групи.

Найважливішою задачею рекомендаційної системи є прогнозування рейтингу. В нашій моделі прогнозування рейтингу буде обчислюватися за рахунок поєднання трьох описаних методів: фільтрація на основі змісту, фільтрація на основі сусідства та демографічна фільтрація.

Тобто, при побудові рекомендацій будемо використовувати 3 методи окремо, для підвищення точності рекомендацій. Кожен з методів буде обчислювати прогнозований рейтинг незалежно від інших. Після чого, отримані результати необхідно лінійно об'єднати, окремо враховуючи внесок результатів кожного методу в результуючому рейтингу. Наприклад, якщо користувач u вже оцінив набір елементів, схожих на елемент s , тоді доцільно збільшити вагу результату, отриманого за допомогою фільтрації за вмістом. Або навпаки, якщо користувач u має велику кількість сусідів, що оцінили елемент s , то визначаючим буде результат фільтрації на основі сусідства. У випадку “холодного старту” доцільно використовувати демографічну фільтрацію. Для обчислення міри впливу результату кожного з методів, введемо ваговий коефіцієнт для кожного методу. Результуючим прогнозованим рейтингом, буде рейтинг, що обчислюється за допомогою наступної формули:

$$r_{us} = \frac{\alpha r_{us}^{DF} + \beta r_{us}^{CB} + \gamma r_{us}^{CF}}{\alpha + \beta + \gamma}, \quad (2.17)$$

де r_{us}^{DF} - прогнозований рейтинг елементу s для користувача u , отриманий за допомогою демографічної фільтрації, r_{us}^{CB} - прогнозований рейтинг елементу s для користувача u , отриманий за допомогою колаборативної фільтрації, r_{us}^{CF} - прогнозований рейтинг елементу s для користувача u , отриманий за допомогою фільтрації на основі вмісту. α, β, γ - відповідні вагові коефіцієнти.

Вклад результатів роботи кожного алгоритму у фінальну оцінку буде залежити від значення відповідного вагового коефіцієнту. Зі зростанням кількості оцінок користувачів в системі, необхідно збільшувати значення вагового

коефіцієнта γ , так як у цьому випадку більш доцільним є застосування колаборативної фільтрації, і навпаки – при невеликій кількості оцінок користувачів в середині системи, або у випадку проблеми “холодного старту” – необхідно збільшувати вклад результатів роботи демографічної фільтрації та фільтрації за вмістом. Для обчислення значень вагових коефіцієнтів необхідно застосувати деяку функцію $f(x)$, де x - кількість оцінок користувачів в середині системи, яка дає значення 1 для великих значень аргументу та 0 для малих значень аргументу. Цим умовам чудово відповідає функція сигмоїда, яка задається за допомогою логістичної рівності:

$$S(x) = \frac{1}{1 + e^{-x}} = \frac{e^x}{e^x + 1} \quad (2.18)$$

Дана функція має наступний вигляд (рисунок 2.4):

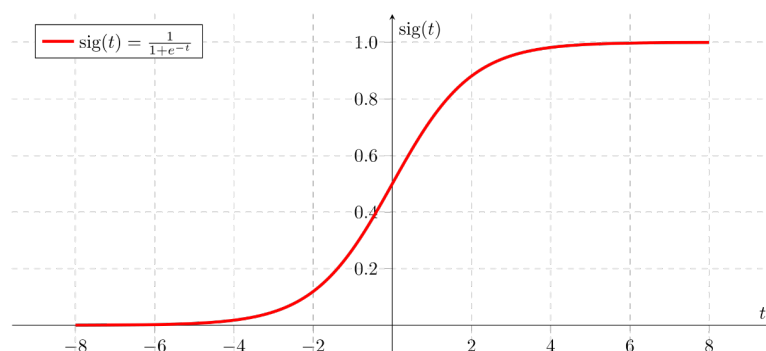


Рисунок 2.4 – Графік функції $S(x)$

Використовуючи цю функцію, задамо вагові коефіцієнти за допомогою наступної системи рівнянь:

$$\begin{cases} \gamma = \frac{1}{1 + e^{-\frac{x}{2}}} \\ \alpha = \beta = 1 - \frac{1}{1 + e^{-\frac{x}{2}}} \end{cases} \quad (2.19)$$

Де x – кількість оцінок користувачів в середині системи. Тобто значення параметрів α, β, γ змінюються динамічно в залежності від доступних оцінок.

2.5 Алгоритм застосування модифікованого підходу

Опишемо алгоритм застосування модифікованого підходу у вигляді 14 кроків:

Крок 1. ПЕРЕЙТИ до демографічної фільтрації.

Крок 2. ОБЧИСЛИТИ вагові коефіцієнти для кожного користувача, на основі подібності демографічних даних (вік, стать, національність і т.д.) з заданим користувачем використовуючи міру схожості (2.15).

Крок 3. ВИБРАТИ зі списку користувачів, які є найбільш наближеними до даного користувача, групу визначеного розміру.

Крок 4. ОБЧИСЛИТИ прогнозований рейтинг елемента на основі оцінок користувачів з обраної групи (2.10).

Крок 5. ПЕРЕЙТИ до фільтрації за вмістом.

Крок 6. ОБЧИСЛИТИ вагові коефіцієнти для кожного елемента, на основі подібності властивостей елемента (жанр, рік видання, кількість сторінок та ін.) з заданим елементом використовуючи міру схожості (2.16).

Крок 7. ВИБРАТИ зі списку елементів, групу елементів, які є найбільш подібними до даного, на основі вагових коефіцієнтів.

Крок 8. ОБЧИСЛИТИ прогнозований рейтинг елемента на основі оцінок подібних елементів (2.7).

Крок 9. ПЕРЕЙТИ до колаборативної фільтрації.

Крок 10. ОБЧИСЛИТИ вагові коефіцієнти для кожного користувача, на основі подібності їх вподобань з вподобаннями заданого користувача, використовуючи міру схожості (2.15).

Крок 11. ВИБРАТИ зі списку користувачів, які є найбільш наближеними до даного користувача, групу визначеного розміру.

Крок 12. ОБЧИСЛИТИ прогнозований рейтинг елемента на основі оцінок користувачів з обраної групи (2.10).

Крок 13. ОБЧИСЛИТИ вагові коефіцієнти для кожного з методів фільтрації (2.19).

Крок 14. ОБЧИСЛИТИ результуючий прогнозований рейтинг елементу (2.18).

На цьому робота алгоритму завершується.

2.6 Результати досліджень ефективності методу

При проведенні експериментів використовувався набір даних с відкритим доступом Goodbooks 10K. Даний набір даних включає в себе детальну інформацію про книги (назва, рік видання, жанр і т.д.), авторів (ім'я, роки життя, рідна мова і т.д.), а також публічну інформацію про користувачів та їх оцінки книгам. Але для проведення експериментів ми можемо обмежитися лише використанням трьох інформаційних полів:

- ідентифікатор книги;
- ідентифікатор користувача;
- оцінка книги користувачем.

Даний набір даних містить в собі інформацію про 10000 книг та оцінки 50000 користувачів в наступному вигляді (таблиця 2.1):

Таблиця 2.1 – Структура експериментальних даних

Ідентифікатор книги	Ідентифікатор користувача	Оцінка
1	1	5
245	1	2
5246	4	3
245	4	1
332	4	5
1	4	4
...

Оцінки являють собою числа в інтервалі від 1 до 5. В системі, що розроблюється, оцінки будуть надаватися у вигляді булевого маркеру – сподобалось/не сподобалось (позитивна/негативна оцінка), тому застосуємо для експериментальних даних наступне перетворення:

$$r'_{us} = \begin{cases} 0, \text{ якщо } r_{us} \in [1,2], r_{us} \in N \\ 1, \text{ якщо } r_{us} \in [3,5], r_{us} \in N \end{cases} \quad (2.20)$$

Тобто, вважатимемо, що користувачу u не сподобався елемент s , якщо його оцінка $r_{us} \in [1,2]$ і навпаки – сподобався, якщо його оцінка $r_{us} \in [3,5]$. Тобто, отримаємо дані наступного вигляду (таблиця 2.2):

Таблиця 2.2 – Структура експериментальних даних після перетворення

Ідентифікатор книги	Ідентифікатор користувача	Оцінка
1	1	1
245	1	0
5246	4	1
245	4	0
332	4	1
1	4	1
...

Далі поділимо дані на 2 частини у наступному співвідношенні (25%/75%):

- тестові дані (25%);
- дані для обчислення (75%).

Тобто необхідно спрогнозувати оцінки елементів користувачами на основі даних для обчислення (75%), після чого порівняти отримані оцінки з тестовими (25%). Таке порівняння надає нам розуміння про ефективність алгоритму, що використовується, а також показує релевантність прогнозованих оцінок.

В якості оцінок ефективності алгоритмів будемо використовувати наступні міри релевантності [27]:

- точність;
- повнота;

- F – міра (F1 score).

Дані міри обчислюються наступним чином:

$$P = \frac{t_p}{t_p + f_p} \quad (2.20)$$

$$R = \frac{t_p}{t_p + f_n} \quad (2.21)$$

$$F = 2 \cdot \frac{P \cdot R}{P + R}, \quad (2.22)$$

де P – точність, R – повнота, F – F -міра, t_p – істинно-позитивні спрацювання (true-positive), f_p – хибно-позитивні спрацювання (false-positive), f_n – хибно-негативні спрацювання (false-negative). В цьому випадку:

- істинно-позитивні спрацювання (true-positive) – оцінки що, є позитивними у тестових даних та у результатах роботи алгоритму;
- хибно-позитивні спрацювання (false-positive) – оцінки що, є негативними у тестових даних але є позитивними у результатах роботи алгоритму;
- хибно-негативні спрацювання (false-negative) – оцінки що, є позитивними у тестових даних але є негативними у результатах роботи алгоритму.

Експерименти будемо проводити використовуючи різні комбінації вищезазначених алгоритмів:

- колаборативна фільтрація (**CF**);
- фільтрація на основі вмісту (**CB**);
- демографічна фільтрація (**DF**);
- **CB + CF**;
- **CF + DF**;
- **DF + CB**;
- модифікований підхід (**CF+CB+DF**).

Результати експериментів зображені у вигляді таблиці 2.3 та рисунку 2.5:

Таблиця 2.3 – Результати експериментів

Алгоритм	Точність	Повнота	F-міра
CF	0.621	0.54	0.577
CB	0.719	0.69	0.704
DF	0.651	0.63	0.64
CB+CF	0.639	0.641	0.64
CF+DF	0.609	0.61	0.609
DF+CB	0.65	0.659	0.654
Модифікований підхід	0.722	0.7	0.71

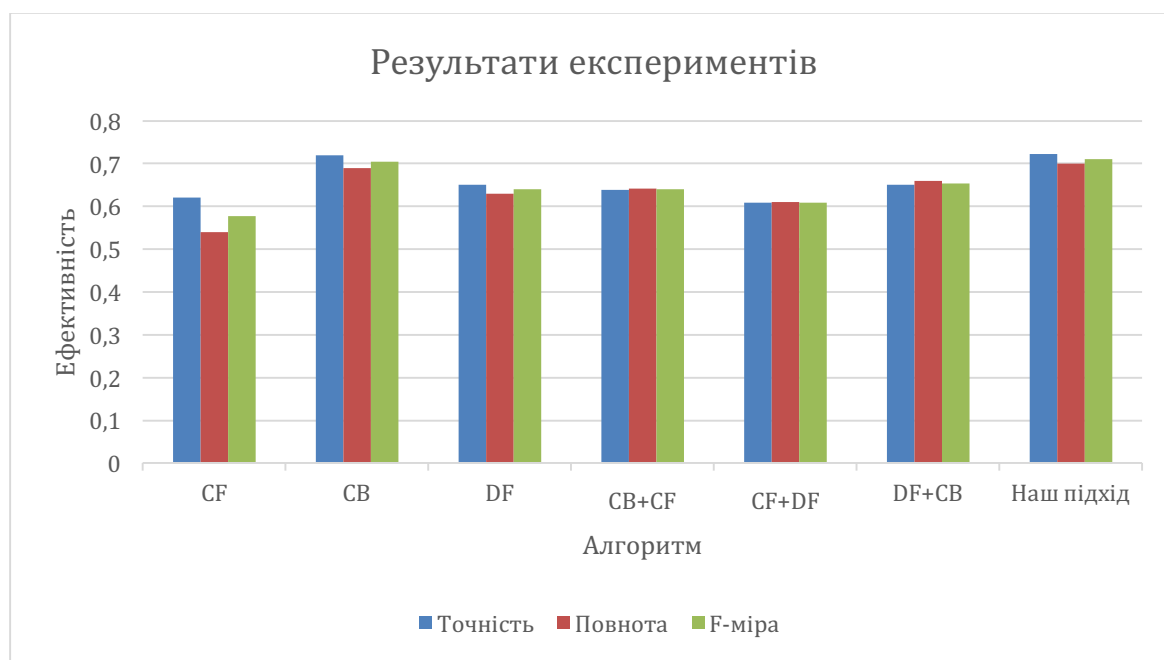


Рисунок 2.5 – Порівняння підходів

Як видно з результатів експериментів – наш підхід дає досить точні прогнози, в порівнянні з іншими підходами (точність - 0.722, повнота – 0.7, F – міра – 0.71).

На основі отриманих даних можна зробити висновок, що колаборативна фільтрація робить значний вклад в кінцевий результат, що відповідно збільшує значення γ та зменшує два інших вагових коефіцієнти (α і β – формула 2.19). Тим не менш, у разі виникнення проблеми холодного старту, демографічна фільтрація буде сприяти зменшенню значення параметра γ , що відповідно зменшить вплив

результатів колаборативної фільтрації на фінальний результат, так як використання колаборативної фільтрації буде неефективним. Розроблена модифікація алгоритму дозволяє обчислювати прогнозовані рейтинги для всіх користувачів, а також у випадку виникнення проблеми холодного старту, використовуючи одну формулу, яка об'єднує параметри, що приймають значення у відповідності з фактичним станом користувача з точки зору кількості доступних оцінок. Динамічний спосіб обчислення параметрів α , β і γ робить наш підхід більш гнучким та практичним, в порівнянні з іншими.

Висновок до розділу

В результаті виконання даного розділу був огляд моделей та методів, які використовуються під час формування рекомендацій, а саме: колаборативна фільтрація та фільтрація за вмістом. Також було розглянуто гібридний підхід, а також описана його модифікація з використанням колаборативної та демографічної фільтрації, а також фільтрації за вмістом. Була проведена серія експериментів, в результаті якої була доведена ефективність модифікованого алгоритму.

3 ОПИС ПРОГРАМНОГО ТА ТЕХНІЧНОГО ЗАБЕЗПЕЧЕННЯ

3.1 Засоби розробки

При створенні інформаційної системи були використані наступні засоби розробки:

Ruby [28] – об'єктно-орієнтована, інтерпретована мова програмування з відкритим вихідним кодом, розроблена Юкіхіро Мацумото який вибрав в якості назви мови – назву дорогоцінного каменю (рубіну), що підкреслити її цінність. Ruby розроблений для простої, гнучкої та розширюваної розробки програмних продуктів. Розроблений на Linux, Ruby є сумісним з більшістю операційних систем: більшість UNIX-платформ, DOS, Windows, Macintosh, BeOS, OS/2 та ін. Синтаксис мови є досить простим, в порівнянні з іншими, що робить код зручним для читання та розуміння.

Ruby on Rails [29] — повноцінний, багаторівневий фреймворк для побудови веб-застосунків, що використовують бази даних, та базується на моделі Model-View-Controller (Model-View-Controller, MVC). Динамічний AJAX-інтерфейс, обробка запитів і видача даних в контролерах, предметна область, відображена в базі даних, - для всього цього Rails надає однорідне середовище розробки на Ruby. Все, що необхідно для початку - база даних і веб-сервер. Основна різниця між Ruby on Rails та іншими фреймворками для розробки полягає в швидкості та простоті використання, що значно підвищує зручність роботи з програмним середовищем. Зміни, внесені до програм, негайно застосовуються, уникаючи тимчасових процедур, які зазвичай пов'язані з циклом веб-розробки.

Rails складається з декількох компонентів, крім самого Ruby, у тому числі:

- Active Record – об'єктно-реляційний шар відображення;
- Action Pack – менеджер контролерів і функцій перегляду;
- Action Mailer – обробник електронної пошти;
- Action Job – сервіс для роботи з відкладеними задачами;
- Action Web – сервіс, для роботи з web-налаштуваннями;

- Action View – сервіс для роботи з HTML-шаблонами;
- Action Storage – сервіс для роботи з файловими вкладеннями користувача;
- Action Cable – сервіс для взаємодії з web-sockets.

Rails може працювати на більшості веб-серверів, які підтримують CGI, та більшістю систем керування базами даних: MySQL, PostgreSQL, SQLite, SQL Server, DB2 та Oracle. Винайдений Девідом Хейнемесром, Ruby On Rails був розроблений як проект з відкритим кодом та вільним розповсюдженням.

HTML [30] – мова розмітки HyperText, відноситься до мови розмітки для розробки веб-сторінок. Це стандарт, який слугує для проектування вмісту веб-сторінок, визначає базову структуру та код (називається HTML-кодом) для визначення вмісту веб-сторінки, наприклад, тексту, зображення, відео, ігри, серед інших. Це цей стандарт відповідає консорціум World Wide Web (W3C) або WWW Consortium, організація, що спеціалізується на стандартизації практично всіх технологій, пов'язаних із Інтернетом. Вважається, що єдина веб-мова є вирішальним фактором для розвитку та розширення Всесвітньої павутини (WWW). Це стандарт, який був застосований для відображення веб-сторінок і прийнятий всіма поточними браузерами.

CSS [31] — це графічна мова дизайну, для налаштування та встановлення візуального дизайну веб-документів та користувацьких інтерфейсів, написаних у HTML або XHTML; мова може бути застосована до будь-якого XML-документа, включаючи XHTML, SVG, XUL, RSS тощо. Це також дозволяє застосовувати не візуальні стилі, такі як слухові таблиці стилів.

Поряд з HTML та JavaScript, CSS - це технологія, що використовується багатьма веб-сайтами для створення візуально привабливих сторінок, користувацьких інтерфейсів для веб-програм і графічних інтерфейсів для багатьох мобільних додатків.

CSS призначений, перш за все, для розділення вмісту. Завдяки такому поділу можна досягти підвищення доступності документа, надати велику гнучкість і

контроль у специфікації презентаційних характеристик та дозволяє декільком HTML документам поділяти один і той же стиль, використовуючи одну таблицю стилів, розділену в .css-файлі, що зменшує складність і уникає повторення коду в структурі документа.

JavaScript [32] — інтерпретована мова програмування, діалект стандарту ECMAScript. Вона визначається як об'єктно-орієнтована, заснована на прототипах, імперативна, слабо типізована та динамічна.

JavaScript в основному використовується в клієнтській формі, реалізований як частина веб-браузера, що дозволяє вдосконалювати інтерфейс користувача та використовувати динамічні веб-сторінки, хоча на серверній стороні може також використовуватися форма JavaScript (Server-side JavaScript або SSJS). Його використовують не лише в Інтернет додатках, наприклад, у генеруванні PDF-документів, настільних додатках (переважно віджетах) та ін.

З 2012 року всі сучасні браузери повністю підтримують ECMAScript 5.1, версію JavaScript. Старі браузери підтримують щонайменше ECMAScript 3.

JavaScript був розроблений з синтаксисом, подібним до C, хоча він застосовує імена та конвенції мови програмування Java. Однак, Java та JavaScript мають різну семантику та цілі.

Всі сучасні браузери інтерпретують інтегрований код JavaScript на веб-сторінках.

PostgreSQL [33] — це система керування базами даних із відкритим вихідним кодом, опублікована під ліцензією PostgreSQL, подібна до BSD або MIT.

Як і багато інших проектів з відкритим кодом, розробка PostgreSQL не керується компанією або людиною, але її керує спільнота розробників, які працюють в безкорисливих, альтруїстичних, вільних цілях або підтримуються комерційними організаціями. Ця спільнота називається PGDG (Global Development Group PostgreSQL).

PostgreSQL багато в чому відповідає стандарту SQL 2011. Більшість функцій доступні і ведуть себе так, як це визначено згідно з стандартом. PostgreSQL повністю сумісний з ACID (включаючи мову визначення даних) та підтримує розширювані типи даних, оператори, функції та агрегати. Незважаючи на те, що спільнота розробників дуже тісно дотримується стандарту SQL, існує ще ряд функціональних можливостей PostgreSQL із зазначенням у документації для кожного ресурсу, незалежно від того, чи відповідає він стандарту SQL, чи це унікальний функціонал. Крім того, PostgreSQL має широкий спектр додаткових вдосконалень.

3.2 Архітектура програмного забезпечення

3.2.1 Діаграма класів

При розробці даного програмного забезпечення використовувалась ORM ActiveRecord. ORM - технологія програмування, яка зв'язує бази даних з концепціями об'єктно-орієнтованих мов програмування, створюючи «віртуальну об'єктну базу даних». У цьому випадку — таблиці БД відповідає певний клас, рядок таблиці БД — це екземпляр певного класу, стовпчик таблиці БД — атрибут певного класу. Це дуже зручний спосіб взаємодії з БД, при цьому вся бізнес-логіка зберігається у моделях. Саме тому в цьому випадку діаграма класів дуже схожа на ER-діаграму (рисунок 3.1 та додаток А):

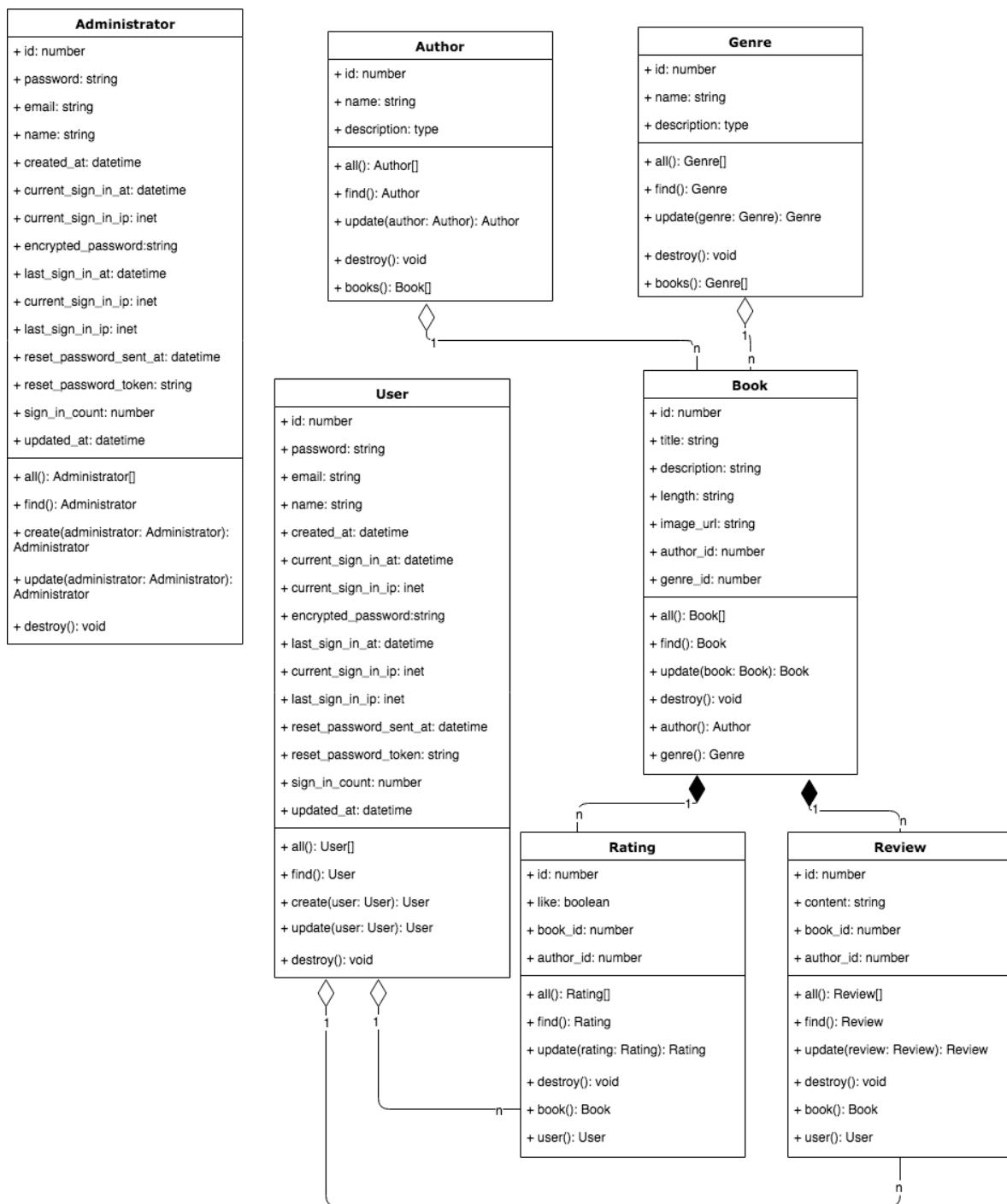


Рисунок 3.1 –Діаграма класів

3.2.2 Діаграма послідовності

Відповідальність кожного з компонентів системи описана у таблиці 3.1:

Таблиця 3.1– Відповідальність класів

Клас	Відповідальність
Користувач	Перегляд каталогу книг, оцінка книг, написання відгуків
Система	Клас верхнього рівня в системі класу ERP. Виконує роль зв'язуючого компоненту між користувачем, реляційною БД та сховищем Redis. Зберігає в собі всю логіку.
Сервер сховища даних “ключ-значення” (Redis)	Зберігає в собі дані про оцінки користувачів. Виконує обробку оцінок користувачів для надання рекомендацій.
Сервер реляційної БД (PostgreSQL)	Сервер БД надає клієнтам доступ до таблиць. Оброблює запити що надходять з серверу.

Зобразимо схему роботи системи у вигляді діаграми послідовності, зображеної на рисунку 3.2 та у додатку А:

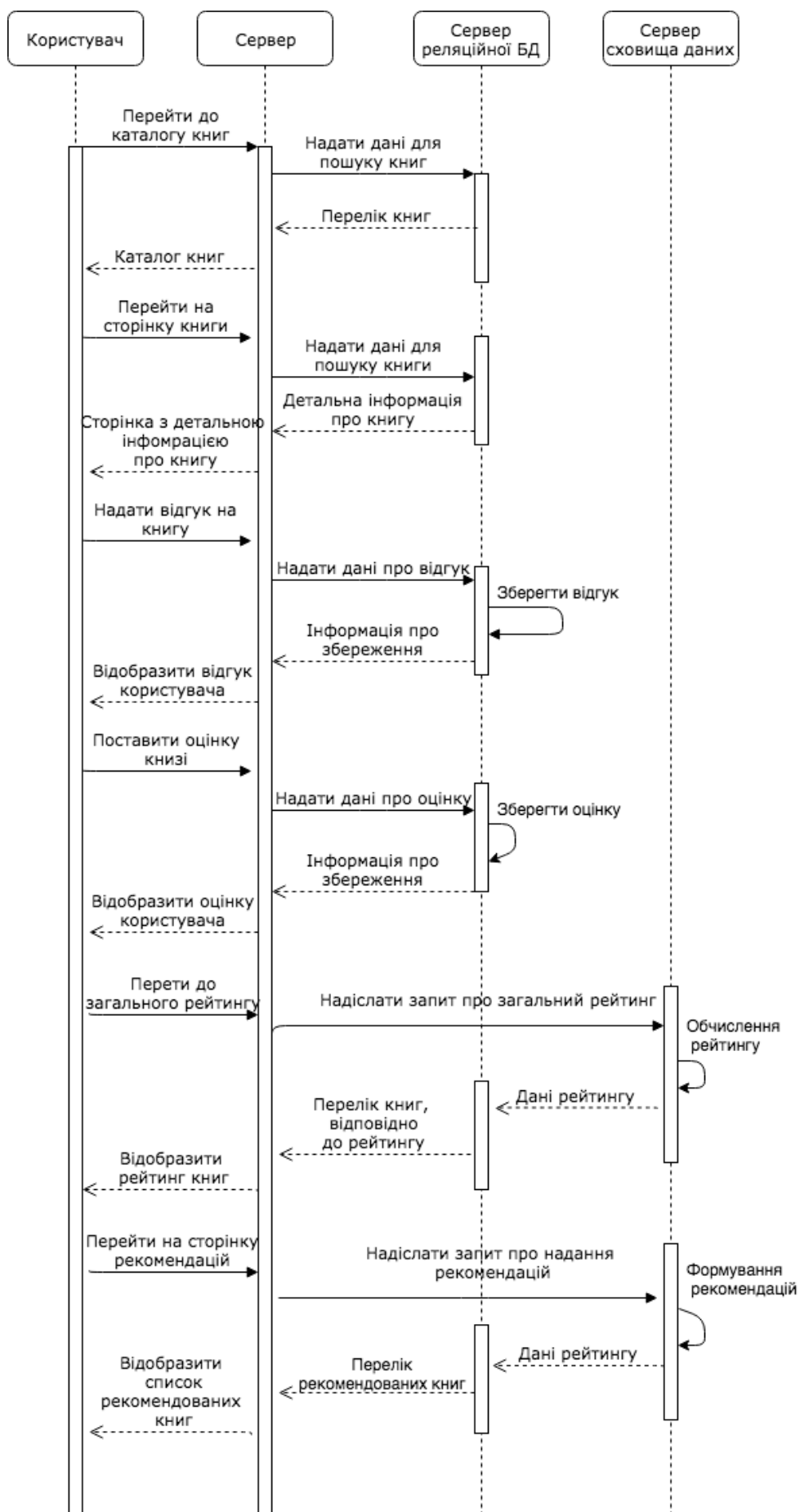


Рисунок 3.2 –Діаграма послідовності

3.2.3 Діаграма компонентів

При розробці даного програмного забезпечення було використано архітектурний шаблон MVC. Модель–вигляд–контролер — архітектурний шаблон, який використовується під час проектування та розробки програмного забезпечення.

Цей шаблон передбачає поділ системи на три взаємопов'язані частини: модель даних, вигляд (інтерфейс користувача) та модуль керування. Застосовується для відокремлення даних (моделі) від інтерфейсу користувача (вигляду) так, щоб зміни інтерфейсу користувача мінімально впливали на роботу з даними, а зміни в моделі даних могли здійснюватися без змін інтерфейсу користувача. Тому діаграма компонентів має наступний вигляд (рисунок 3.3 та додаток А):

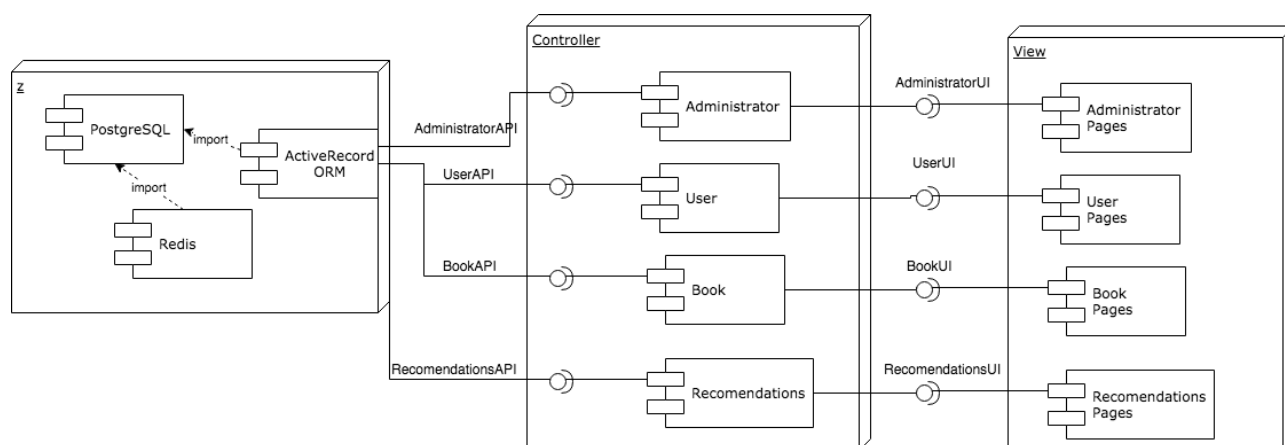


Рисунок 3.3 –Діаграма компонентів

3.3 Інструкція користувача

Для взаємодії з системою користувачу необхідно виконати авторизацію, заповнивши наступну форму входу або зареєструватися у системі (рисунок 3.4):

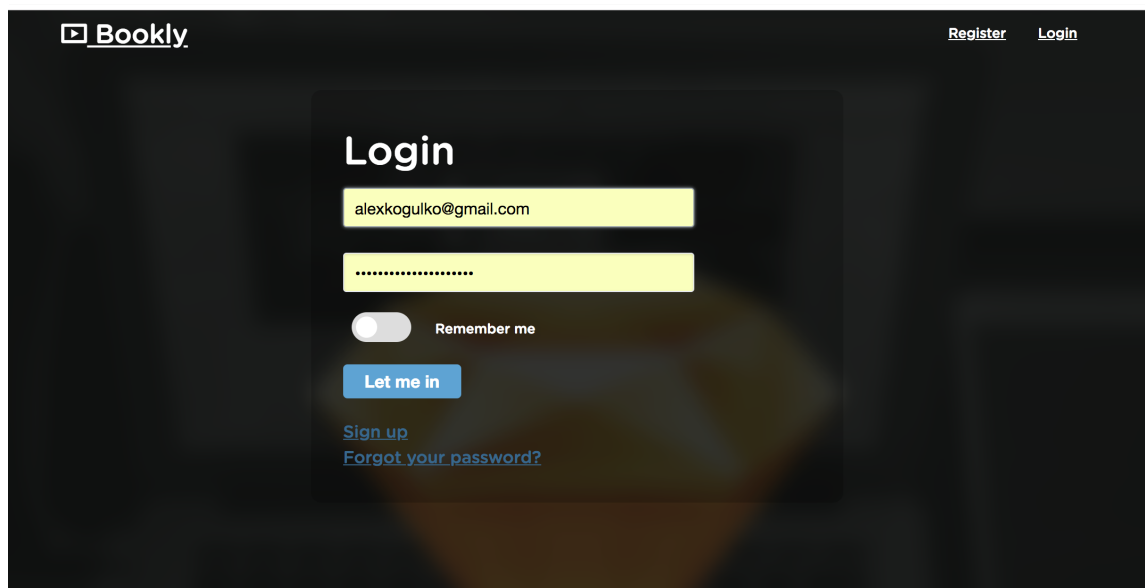


Рисунок 3.4 –Форма входу

Після чого користувач має можливість переглядати каталог книг, який має наступний вигляд (рисунок 3.5):

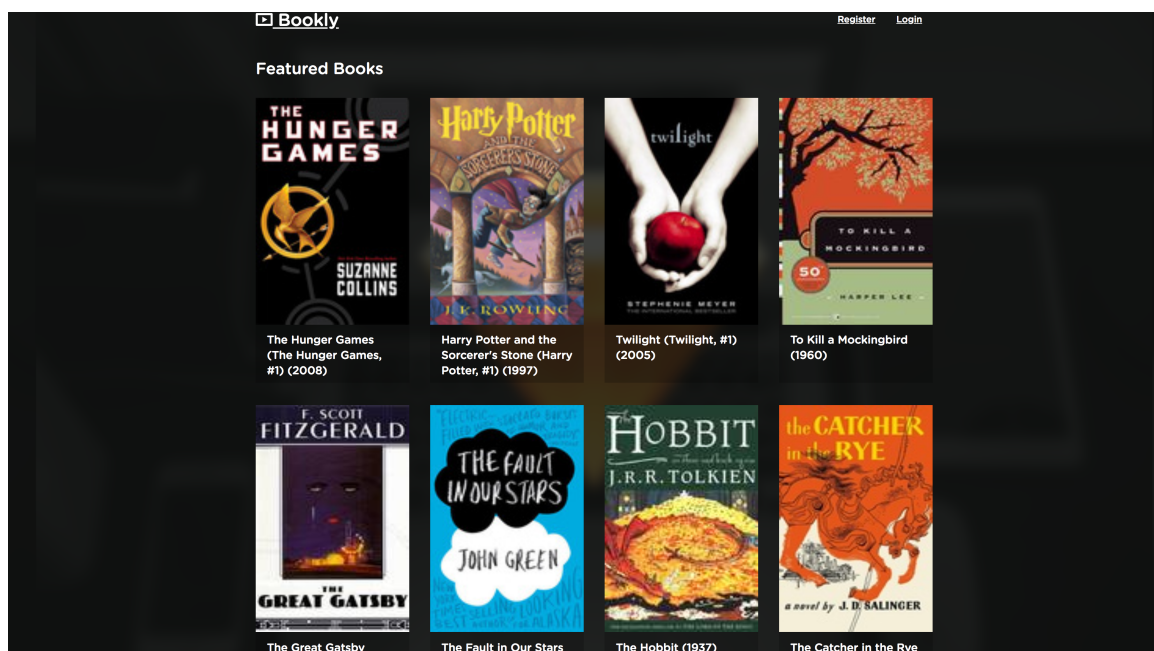


Рисунок 3.5 –Каталог книг

Кожен елемент каталог являє собою окрему книгу. Для кожної книги існує сторінка з детальною інформацією. На ній відображаються основні відомості про книгу, а також відгуки інших користувачів та кнопки, за допомогою яких можна поставити оцінку кожній книзі (рисунок 3.6):

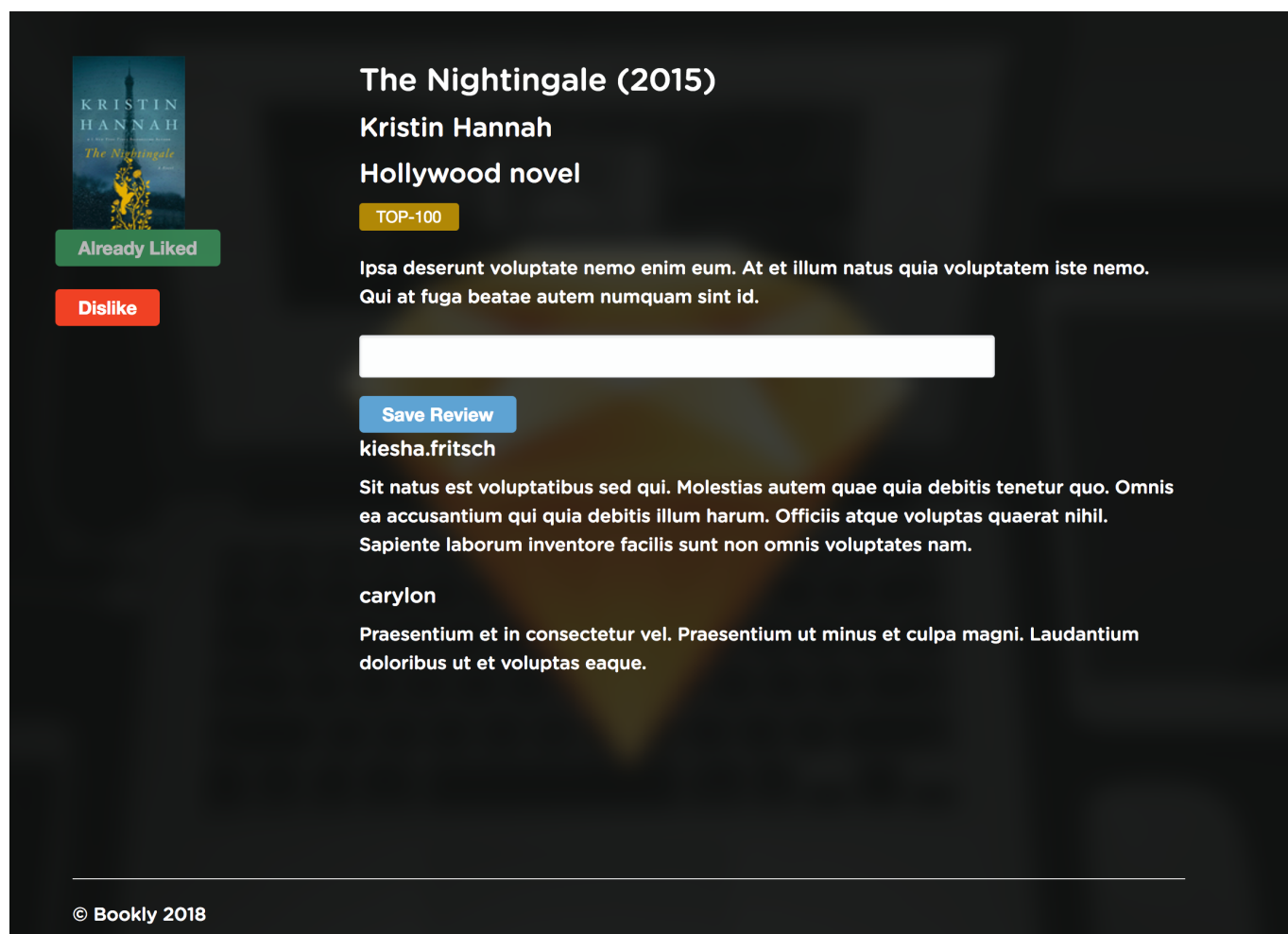


Рисунок 3.6 –Детальна інформація про книгу

Також користувач має можливість залишити власний відгук на книгу заповнивши спеціальну форму.

Зареєстрований користувач має можливість переглянути список книг, сформований на основі особистих рекомендацій (рисунок 3.7), а також - список книг, які є найвище оціненими всіма користувачами системи (рисунок 3.8):

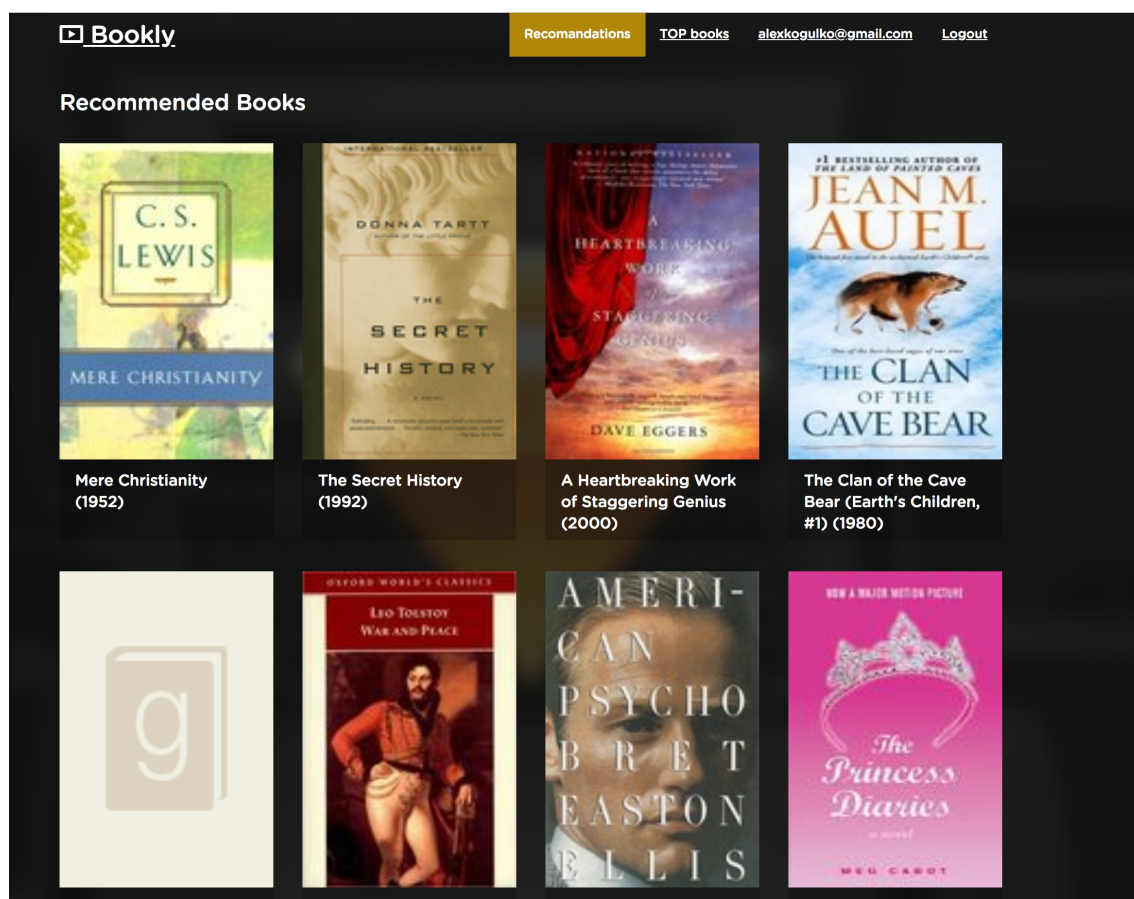


Рисунок 3.7 –Список рекомендованих книг

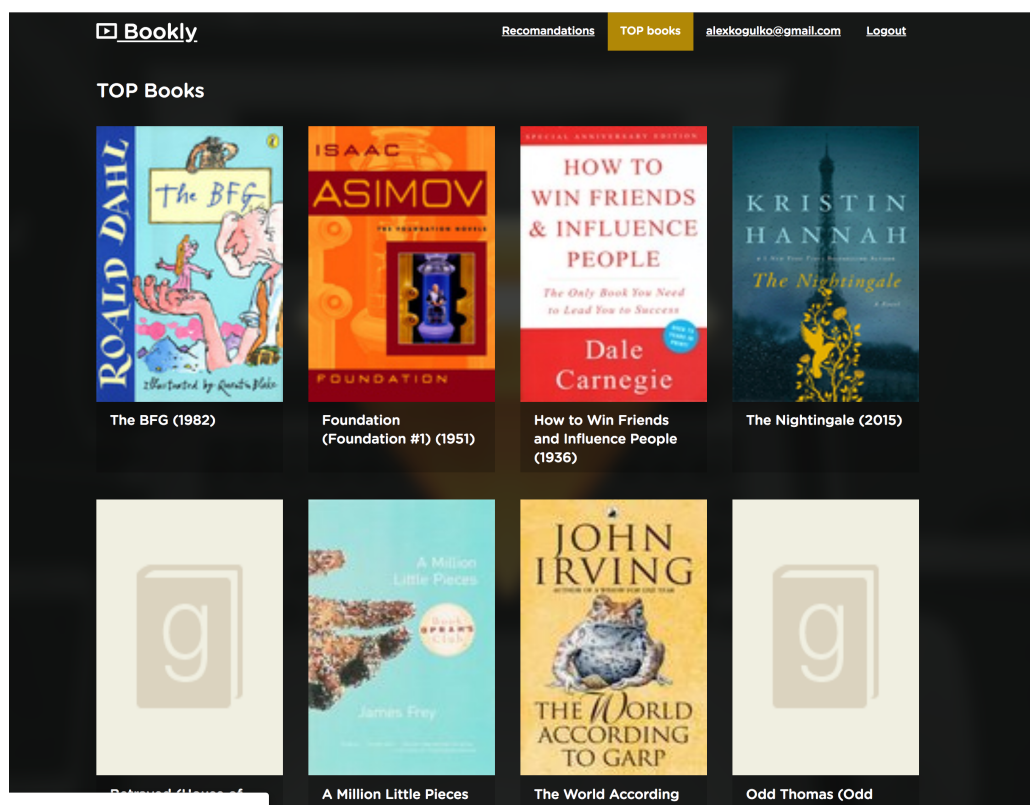
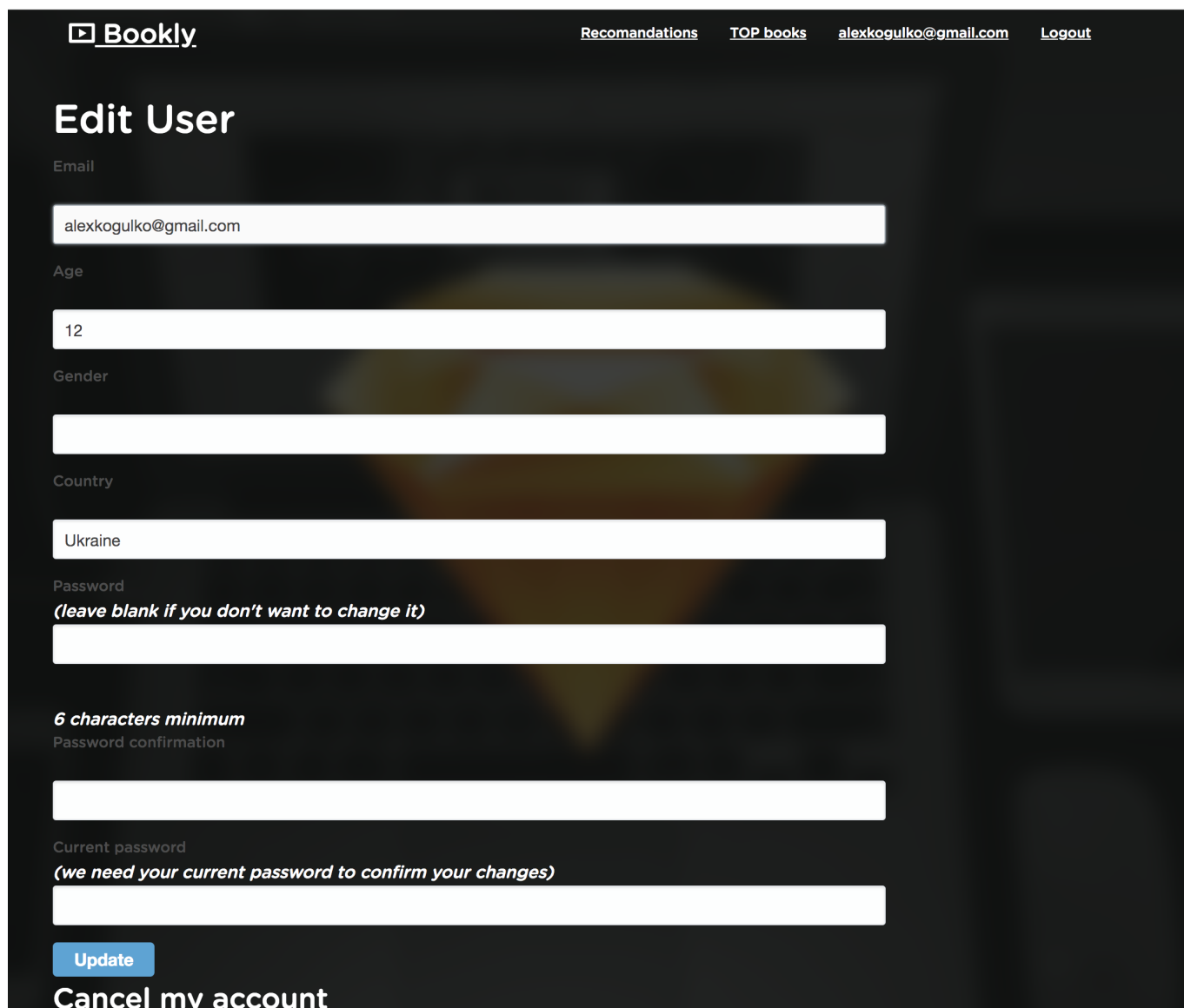


Рисунок 3.8 –Список найвище оцінених книг

Також користувач має можливість редагувати особисту інформацію щодо профілю користувача (рисунок 3.9):



The screenshot shows the 'Edit User' page of the 'Bookly' application. The page has a dark background with white text and form fields. At the top, there is a navigation bar with the 'Bookly' logo, links for 'Recommendations', 'TOP books', the user's email 'alexkogulko@gmail.com', and a 'Logout' link. The main heading is 'Edit User'. Below it, there are several form fields: 'Email' (containing 'alexkogulko@gmail.com'), 'Age' (containing '12'), 'Gender' (empty), 'Country' (containing 'Ukraine'), 'Password' (with a note '(leave blank if you don't want to change it)'), 'Password confirmation' (with a note '6 characters minimum'), and 'Current password' (with a note '(we need your current password to confirm your changes)'). At the bottom, there is a blue 'Update' button and a link 'Cancel my account'.

Рисунок 3.9 – Профіль користувача

Для взаємодії адміністратора з системою передбачена спеціальна панель, яка дозволяє зручно вести реєстр книг, авторів та жанрів а також переглядати інформацію про користувачів системи (рисунок 3.10):

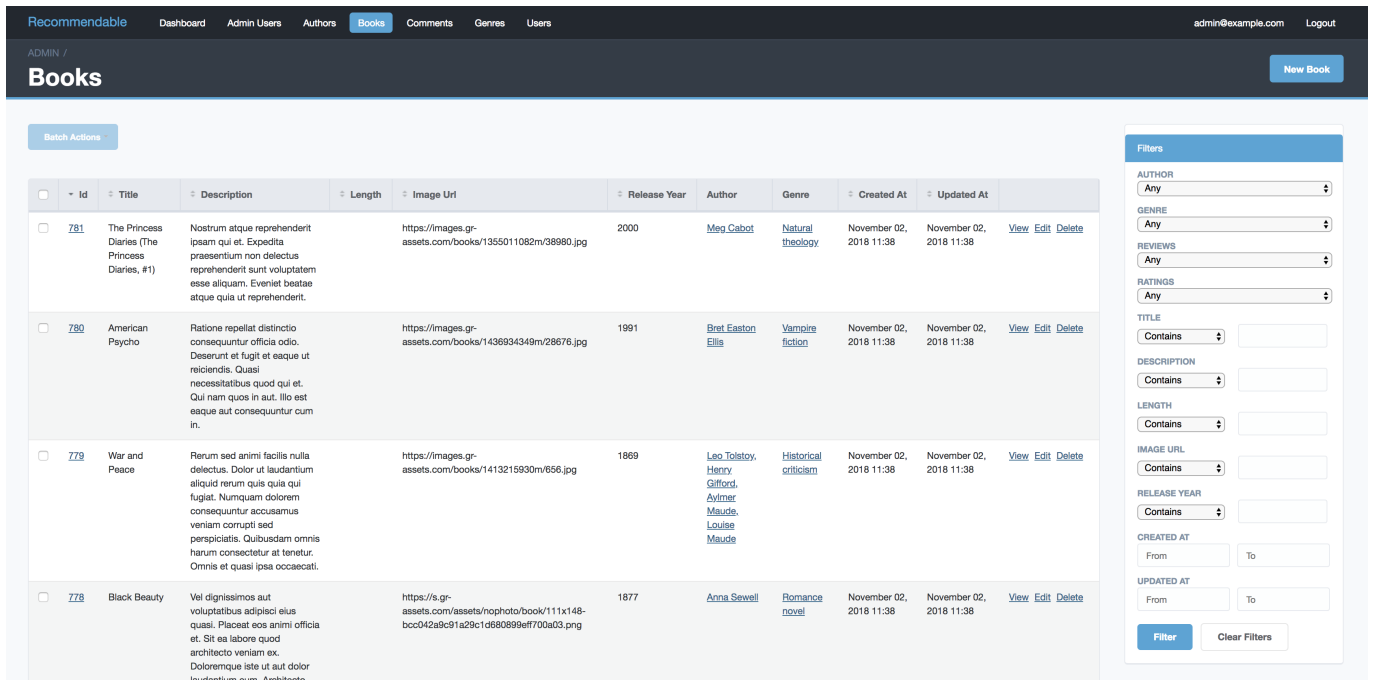


Рисунок 3.10 – Панель адміністратора

3.4 Опис технічного забезпечення

Діаграма розгортання має наступний вигляд (рисунок 3.11 та додаток А):

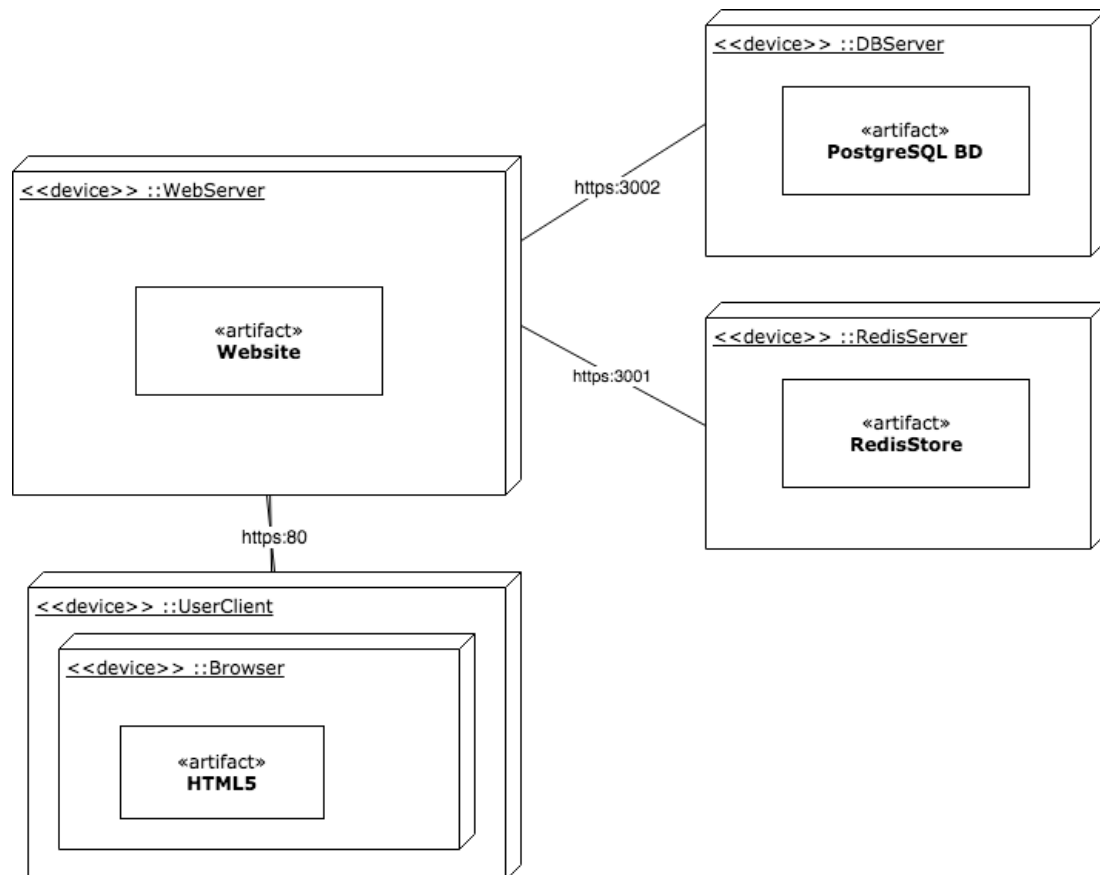


Рисунок 3.11 –Діаграма розгортання

Структура технічних засобів залежить від способу взаємодії користувача з системою, завдань, поставлених до системи, вимог до захищеності, можливості інтегрування та ресурсів, доступних до застосування.

Для коректної роботи серверної частини системи, що розробляється, рекомендовано використовувати сервер, з наступними характеристиками:

- 4-ядерний процесор з частотою не менш ніж 2.0 ГГц;
- оперативна-пам'ять формату DDR5 та об'ємом не менш ніж 16 ГБ;
- тверdotілий накопичувач (SSD) об'ємом не менш, ніж 256 Гб;
- безперервний доступ до мережі.

Для комфортної взаємодії користувача з системою, рекомендовано використовувати персональний комп'ютер, що відповідає наступних характеристикам:

- 2-ядерний процесор з частотою не менш ніж 1.6 ГГц;
- оперативна-пам'ять формату об'ємом не менш ніж 4 ГБ;
- безперервний доступ до мережі Інтернет;
- остання версія браузеру Google Chrome [34].

Висновок до розділу

В результаті виконання даного розділу був виконаний опис програмного та технічного забезпечення яке було використано під час розробки системи. В результаті були сформовані діаграми класів, послідовності, компонентів та розгортання які характеризують програмну та технічну архітектуру системи. Також була складена інструкція користувача, яка описує основні варіанти використання системи користувачем.

4 РОЗРОБКА СТАРТАП-ПРОЕКТУ

4.1 Опис ідеї стартап-проекту

Основна ідея стартап проекту полягає у наступному (таблиця 4.1):

Таблиця 4.1. Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Сервіс, який дозволяє користувачам переглядати інформацію про книги, переглядати рейтинги, надавати відгуки та отримувати персональні рекомендації.	1. Перегляд каталогу книг	Отримання загальної інформації про книги
	2. Перегляд детальної інформації про книги, жанри та авторів	Отримання детальної інформації про книгу/автора, що зацікавив користувача.
	3. Перегляд оцінок книг, загального рейтингу та відгуків інших користувачів	Отримання попереднього враження про книгу
	4. Отримання рекомендацій	Отримання персональних рекомендацій на основі власних побажань

4.1.1 Аналіз потенційних техніко-економічних переваг ідеї

До основних конкурентів сервісу, що розробляється відносяться:

- Amazon;
- Goodreads;
- LiveLib.ru.

Виконаємо аналіз кожного сервісу у вигляді таблиці 4.2:

Таблиця 4.2. Визначення сильних, слабких та нейтральних характеристик ідеї проекту

п / п	Техніко-економічні характеристики ідеї	(потенційні) товари/концепції конкурентів				(слабка сторона)	(нейтральна сторона)	(сильна сторона)
		Мій проект	Amazon	Goodreads	LiveLib.ru			
1	Витрати на обслуговування	Близько 50\$ на місяць	Конфіденційна інформація	Конфіденційна інформація	Конфіденційна інформація			+
2	Витрати на використання	Близько 50\$ на місяць	Конфіденційна інформація	Конфіденційна інформація	Конфіденційна інформація			+
3	Надійність	Висока, за рахунок простоти сервісу	Дуже висока	Дуже висока	Висока		+	
4	Масштабованість	Висока	Дуже висока	Дуже висока	Конфіденційна інформація		+	
5	Вартість підтримки	Так як система розробляється в рамках виконання дипломного проекту – підтримка на перших етапах є безкоштовною	Дуже висока	Висока	Висока			+
6	Зручність	Висока за рахунок мінімалістичного дизайну	Висока	Висока	Висока		+	

Продовження таблиці 4.2

п/п	Техніко-економічні характеристик і ідеї	(потенційні) товари/концепції конкурентів				(слабка сторона)	(нейтральна сторона)	(сильна сторона)
		Мій проект	Amazon	Goodreads	LiveLib.ru			
7	Поріг входу	Низький	Середній	Середній	Середній			+
8	Відповідність сучасним вимогам UX-дизайну	Висока за рахунок мінімалістичного інтерфейсу	Висока	Висока	Висока		+	
9	Відповідність сучасним вимогам UI-дизайну	Відповідність базовим вимогам	Низька. Застарілий дизайн	Низька. Застарілий дизайн	Низька. Застарілий дизайн			+
10	Рівень захисту інформації	Висока. Система не зберігає мінімальну кількість особистих даних.	Дуже висока	Дуже висока	Висока		+	
1	Відмовостійкість	Висока, за рахунок простоти архітектури	Дуже висока	Дуже висока	Дуже висока		+	

4.2 Технологічний аудит ідеї проекту

Технологічна здійсненність проекту описана в таблиці 4.3:

Таблиця 4.3 Технологічна здійсненність ідеї проекту

№ п/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1	Завантаження бази книг	Відкриті електронні бібліотеки книг	Наявна	Доступна

№ п/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
2	Інформаційна система з пошуку книг	Засоби розробки інформаційної системи	Потребує розробки	Доступна за рахунок використання засобів з відкритим вихідним кодом
3	Надання особистих комендацій користувачам	Алгоритми фільтрації та прогнозування	Потребує розробки	Доступна за рахунок використання засобів з відкритим вихідним кодом та відомих алгоритмів
4	Розповсюдження та отримання особистих вражень від книги	Засоби розробки інформаційної системи	Потребує розробки	Доступна за рахунок використання засобів з відкритим вихідним кодом
Обрана технологія реалізації ідеї проекту: Гібридний підхід для формування особистих рекомендацій за рахунок комбінації результатів колаборативної та демографічної фільтрації та фільтрації за вмістом.				

4.3 Аналіз ринкових можливостей запуску стартап-проекту

Характеристика потенційного ринку стартап-проекту, описана в таблиці 4.4:

Таблиця 4.4. Попередня характеристика потенційного ринку стартап-проекту

№ п/п	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од	3
2	Загальний обсяг продаж, грн/ум.од	неможливо встановити
3	Динаміка ринку (якісна оцінка)	спадаюча

№ п/п	Показники стану ринку (найменування)	Характеристика
4	Наявність обмежень для входу (вказати характер обмежень)	Присутні обмеження: 1. Відсутність коштів на рекламу 2. Недовіра з боку користувачів
5	Специфічні вимоги до стандартизації та сертифікації	відсутні
6	Середня норма рентабельності в галузі (або по ринку), %	неможливо встановити

Характеристика потенційних клієнтів стартап-проекту, описана в таблиці 4.5:

Таблиця 4.5. Характеристика потенційних клієнтів стартап-проекту

№ п/п	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
1	Необхідність пошуку цікавих для конкретного користувача книг	Будь який користувач Всесвітньої Мережі	Різниця в поведінці за різними факторами: вік, стать, національність та ін.	Релевантність рекомендованих книг
2.	Можливість поділитися власною думкою та дізнатися думки інших	Будь який користувач Всесвітньої Мережі	Різниця в поведінці за різними факторами: вік, стать, національність та ін.	Модерація відгуків користувачів відповідно до загальноприйнятих нормам моралі та етикету

Основні фактори загроз стартап-проекту, описані в таблиці 4.6:

Таблиця 4.6. Фактори загроз

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
1	Зниження інтересу сучасної молоді до читання книг	Зниження популярності предметної області сервісу - книг	Відсутня
2	Недостатня інформаційна база на початку існування проекту	Недостатня кількість інформації про книги для повноцінного функціонування каталогу та рекомендаційної системи	Залучення відкритих баз даних з інформацією про книги
3	Недостатня популярність сервісу	Відсутність уваги з точки зору користувачів	Рекламна компанія сервісу. Залучення SEO-спеціалістів та маркетологів для популяризації сервісу в мережі Інтернет

Основні фактори можливостей стартап-проекту, описані в таблиці 4.7:

Таблиця 4.7. Фактори можливостей

№ п/п	Фактор	Зміст можливості	Можлива реакція компанії
1	Залучення професійних критиків для написання авторитетних відгуків	Збільшення довіри до сервісу з точки зору користувачів	Виділення коштів на оплату роботи критиків
2	Партнерство з магазинами книг для розповсюдження книг через рекомендаційну систему	Розширення системи. Додаткова комерціалізація	Залучення партнерів в особі магазинів книг

Ступеневий аналіз конкуренції на ринку описаний у таблиці 4.8:

Таблиця 4.8. Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
1. Вказати тип конкуренції - монополія/олігополія/ монополістична/чиста	Олігополія, так як на ринку існує декілька ключових гравців (Amazon, GoodReads)	Орієнтація на ринок країн СНД.
2. За рівнем конкурентної боротьби - локальний/національний/...	Інтернаціональний, так як основні гравці на ринку – інтернаціональні компанії	Вихід на міжнародний ринок, після отримання чіткого становлення на ринку країн СНД
3. За галузевою ознакою - міжгалузева/ внутрішньогалузева	Внутрішньогалузева, так як діяльність систем зосереджена на книгах	Дотримання основних концепцій та принципів функціонування сервісів в даній галузі

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
4. За характером конкурентних переваг - цінова / нецінова	Нецінова, так як сервіс є безкоштовним для використання користувачів	Підтримання високої конкуренції за рахунок унікальних переваг для користувача

Аналіз конкуренції в галузі за М. Портером описаний у таблиці 4.9:

Таблиця 4.9. Аналіз конкуренції в галузі за М. Портером

	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари замітники
Складові аналізу	1) Amazon 2) GoodReads 3) LiveLib.ru	Відсутні	Доступ до системи постачається за разунк мережі Інтернет	Визначити фактори сили споживачів	Рекомендаційні системи компаній-конкурентів
Висновки	Інтенсивність конкурентної боротьби зі сторони прямих конкурентів є низькою	Потенційні конкуренти відсутні	Відсутні постачальники, що диктують умови роботи на ринку	Відсутні клієнти, що диктують умови роботи на ринку	Робота системи обмежується чинним законодавством країни, у яку постачаються послуги

Обґрунтування факторів конкурентоспроможності описано в таблиці 4.10:

Таблиця 4.10. Обґрунтування факторів конкурентоспроможності

№ п/п	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Поєднання рекомендацій від системи з відгуками користувачів	Завдяки цьому користувач зможе співставити відгуки інших користувачів з тим, що на думку системи повинно співпадати з його власними вподобаннями
2	Гібридна система фільтрації	Побудова рекомендацій на основі гібридного методу, що поєднує колаборативну та демографічну фільтрацію а також фільтрацію на основі елементів
3	Побудова рекомендацій на основі бінарної реакції користувачів (сподобалось/не сподобалось) замість використання шкали оцінок	Збільшення релевантності рекомендацій, та цінності кожної окремої оцінки
4	Відсутність плати за використання сервісу	Система є безкоштовною для використання. Комерціалізація буде досягатись за рахунок відображення таргетованої реклами. Як приклад: реклама від книжкових магазинів

№ п/п	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
5	Можливість впровадження нового функціоналу	В майбутньому система може здобути набір нового функціоналу. Інтеграція з книжковими магазинами, аналіз даних з соцмереж, публікація власних книг авторами та ін.

Порівняльний аналіз сильних та слабких сторін системи описано в таблиці 4.11:

Таблиця 4.11. Порівняльний аналіз сильних та слабких сторін системи рекомендацій на основі поведінкової моделі користувача

п/п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з системою рекомендацій на основі поведінкової моделі користувача						
			-3	-2	-1	0	+1	+2	+3
1	Поєднання рекомендацій від системи з відгуками користувачів	15						+	
2	Гібридна система фільтрації	15						+	
3	Побудова рекомендацій на основі бінарної реакції користувачів (сподобалось/не сподобалось) замість використання шкали оцінок	20							+

Продовження таблиці 4.11

№ п/п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з системою рекомендацій на основі поведінкової моделі користувача						
			-3	-2	-1	0	+1	+2	+3
4	Відсутність плати за використання сервісу	15						+	
5	Можливість впровадження нового функціоналу	15							+

SWOT- аналіз стартап-проекту описано в таблиці 4.12:

Таблиця 4.12. SWOT- аналіз стартап-проекту

<p>Сильні сторони:</p> <ul style="list-style-type: none"> – Рекомендаційна система книг – Підсистема для рекламодавців – Система є достить гнучкою для розширення 	<p>Слабкі сторони:</p> <ul style="list-style-type: none"> – Невелика база даних книг – Відсутність достатньої кількості даних про оцінки користувачів, через непопулярність сервісу
<p>Можливості:</p> <ul style="list-style-type: none"> – Оцінка книг користувачами – Надання та перегляд відгуків на книги – Перегляд загального рейтингу книг. – Перегляд особистих рекомендацій. – Можливість впровадження нового функціоналу 	<p>Загрози:</p> <ul style="list-style-type: none"> – Висока конкуренція – Відсутність інтересу сучасної молоді до книг

Альтернативи ринкового впровадження описано в таблиці 4.13:

Таблиця 4.13. Альтернативи ринкового впровадження стартап-проекту

№ п/п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Випуск MVP-версії сервісу в публічний доступ	Ресурси будуть надані інвесторами	2 місяці
2	Покращення візуальної складової сервісу	Відсутня	1 місяць
3	Залучення більшої кількості книг до загального каталогу	Ресурси будуть отримані за рахунок обробки баз даних в відкритим доступом	1 місяць

4.4 Розроблення ринкової стратегії проекту

Вибір цільових груп потенційних споживачів я описано в таблиці 4.14:

Таблиця 4.14. Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Користувачі віком до 14 років будь-якої статі та національності	Недостатня готовність через недостатній інтерес до читання або через можливу відсутність досвіду взаємодії з веб- сервісами	Низький попит	Низька	Низька

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
2	Користувачі віком від 14 до 60 років будь-якої статі та національності	Споживачі повністю готові сприйняти продукт	Середній попит	Низька	Середня
3	Користувачі віком від 60 років будь-якої статі та національності	Недостатня готовність , через можливу відсутність досвіду взаємодії з веб-сервісами	Низький попит	Низька	Низька
Які цільові групи обрано: Користувачі віком від 14 років (2,3 група) будь-якої статі та національності					

Визначення базової стратегії розвитку описано в таблиці 4.15:

Таблиця 4.15. Визначення базової стратегії розвитку

№ п/п	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
1	Стратегія диференціації	Стратегія концентрованого маркетингу	Зосередження лише на конкретному сегменті – рекомендація книг	Стратегія спеціалізації

Визначення базової стратегії конкурентної поведінки описано в таблиці 4.16:

Таблиця 4.16. Визначення базової стратегії конкурентної поведінки

№ п/п	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки
1	Ні	Компанія буде шукати нових споживачів	Ні	Стратегія заняття конкретної ніші

Визначення стратегії позиціонування стартап-проекту описано в таблиці 4.17:

Таблиця 4.17. Визначення стратегії позиціонування

№ п/п	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
1	Релевантність рекомендованих книг	Стратегія спеціалізації	Спеціалізація на рекомендуванні книг	Книги, рекомендації, відгуки
2	Модерація відгуків користувачів відповідно до загальноприйнятих нормам моралі та етикету	Стратегія спеціалізації	Відображення лише доцільних відгуків користувачів	Доцільність, адекватність, вичерпність.

4.5 Розроблення маркетингової програми стартап-проекту

Визначення ключових переваг концепції потенційного товару описано у таблиці 4.18:

Таблиця 4.18. Визначення ключових переваг концепції потенційного товару

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
1	Необхідність пошуку цікавих для конкретного користувача книг	Отримання особистих рекомендацій книг	Рекомендації формуються на основі специфічних властивостей елементів (книг)
2	Можливість поділитися власною думкою та дізнатися думки інших	Ознайомлення з суспільною думкою про дану книгу	Залучення професійних критиків для написання відгуків

Опис трьох рівнів моделі товару здійснений у вигляді таблиці 4.19:

Таблиця 19. Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові		
I. Товар за задумом	Задоволення потреби пошуку цікавих для конкретного користувача книг за допомогою особистих рекомендацій		
II. Товар у реальному виконанні	Властивості/характеристики	М/Нм	Вр/Тх /Тл/Е/Ор
	Дизайн	Нм	Вр/Е
	Зручність використання	Нм	Вр/Е/Тл
	Релевантність рекомендацій	М	Тх/Тл
	Безпека використання	Нм	Вр/Тх/Тл
	Якість: відповідає міжнародним стандартам інформаційної безпеки BS 7799-1: 2005, ISO/IEC 17799: 2005.		

Рівні товару	Сутність та складові
III. Товар із підкріпленням	Інформацію про товар, та його переваги користувач може отримати за допомогою контекстної реклами.
За рахунок чого потенційний товар буде захищено від копіювання за рахунок захисту ідеї товару	

Визначення меж встановлення ціни описано в таблиці 4.20:

Таблиця 4.20. Визначення меж встановлення ціни

№ п/п	Рівень цін на товари замітники	Рівень цін на товари аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1	-	-	-	Сервіс є безкоштовним для використання

Формування системи збуту описано в таблиці 4.21:

Таблиця 4.21 Формування системи збуту

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1	-	Розміщення інформації про продукт на рекламних майданчиках	Висока	Контекстна реклама

Концепція маркетингових комунікацій описана в таблиці 4.22:

Таблиця 22. Концепція маркетингових комунікацій

№ п/п	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1	Перехід за посиланням з рекламного банеру	Контекстна реклама на інших веб-сервісах	Рекомендації книг	Ознайомити користувача з сервісом за зацікавити у його використанні	Розробка рекламних банерів з інформацією про систему та її основними перевагами

Висновок до розділу

В результаті виконання даного розділу був виконаний опис стартап-проекту. Сформовано основну ідею проекту, здійснений технологічний аудит системи. Проведений аналіз ринку подібних систем та огляд основних конкурентів. Виділено сильні та слабкі сторони системи, що розробляється, а також сформована маркетингова програма проекту.

ВИСНОВКИ

У результаті роботи над магістерською дисертацією було виконано огляд існуючих методів та алгоритмів роботи рекомендаційних систем, а саме - колаборативної фільтрації та фільтрації за вмістом. Розглянуто особливості їх застосування та описані основні принципи роботи даних алгоритмів.

Здійснено порівняльний аналіз різних методів та алгоритмів роботи рекомендаційних систем та визначено випадки, коли доцільно використовувати кожен з алгоритмів, розглянуто «вузькі місця» та основні проблеми, які можуть виникнути у результаті застосування алгоритму.

Розглянуто варіанти застосування гібридного підходу для формування рекомендацій. Описані основні варіанти використання даного підходу.

Сформовано задачу та розроблено модель надання рекомендацій на основі модифікованого гібридного підходу за допомогою комбінування результатів роботи 3 алгоритмів (колаборативна фільтрація, демографічна фільтрація, фільтрація за вмістом).

Виконано аналіз результатів роботи алгоритмів, за допомогою серії експериментів, та доведена ефективність використання модифікованого гібридного підходу.

Розроблено прототип рекомендаційної системи книг, на основі сформованого гібридного підходу. Описана архітектура програмного та технічного забезпечення системи, засоби які використовувались під час розробки та сформована інструкція користувача для полегшення взаємодії з системою.

За матеріалами дисертації було опубліковано 2 наукові роботи: 1 стаття та 1 тези доповіді на конференціях [35-36].

Отримані результати мають практичне значення і, адже використовуючи модифікований підхід можна досягнути підвищення релевантності рекомендацій елементів користувачам.

ПЕРЕЛІК ПОСИЛАНЬ

1. J. Bobadilla, F. Ortega, A. Hernando, and A. Guti´errez, “Recommender systems survey,” *Knowledge-Based Systems*, vol. 46, 2013 – P. 109-132.
2. Adomavicius G. Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions / G. Adomavicius, A. Tuzhilin // *IEEE Transactions on Knowledge and Data Engineering* – 2005. – Vol. 17, No6. – P. 734-749.
3. В.В. Куриленко Разработка веб-приложения музыкального аудиостриминга с рекомендательной системой // *НИ ТГУ* – 2017 – P. 7-15.
4. G. Salton *Automatic Text Processing* // Addison-Wesley. – 1989 – 530 pages.
5. Daniar Asanov (n.d.) *Algorithms and Methods in Recommender Systems*, Berlin, Germany: Berlin Institute of Technology – 2011 – P. 1-7.
6. Jones, M. T. (2013, December 12). *Recommender systems. Introduction to approaches and algorithms*. Retrieved November 25, 2017, from <https://www.ibm.com/developerworks/library/os-recommender1/>
7. Jerold Angelus Grundy *Newbrain* // Duct Publishing – 2012 – 120 pages.
8. D. Goldberg *Using Collaborative Filtering to Weave an Information Tapestry* / D. Goldberg, D. Nichols, B.M. Oki, D. Terry // *Comm. ACM*. – 1992. – Vol. 35, No12. – P. 61-70.
9. Gleb Beliakov, Tomasa Calvo and Simon James. *Aggregation of preferences in recommender systems* / Gleb Beliakov and Simon James // *School of Information Technology, Deakin University* – 2008 – P. 705-735.
10. J. Basilico and T. Hofmann. *Unifying collaborative and content-based filtering*. In *Proceedings of the 21th International Conference on Machine Learning*, 2004 – P. 9-16
11. C.C. Aggarwal *Horting Hatches an Egg: A New Graph-Theoretic Approach to Collaborative Filtering* / C.C. Aggarwal, J.L. Wolf, K-L. Wu, P.S. Yu // *Proc. Fifth ACM SIGKDD Int’l Conf. Knowledge Discovery and Data Mining*. – 1999 – P. 201-212.
12. P. Resnick *GroupLens: An Open Architecture for Collaborative Filtering of Netnews* / P. Resnick, N. Iakovou, M. Sushak, P. Bergstrom, J. Riedl // *Proc. 1994 Computer Supported Cooperative Work Conf.* – 1994 – P. 175-186.

13. U. Shardanand Social Information Filtering: Algorithms for Automating “Word of Mouth” / U. Shardanand, P. Maes // Proc. Conf. Human Factors in Computing Systems. – 1995 – P. 210-217.
14. J.S. Breese Empirical Analysis of Predictive Algorithms for Collaborative Filtering / J.S. Breese, D. Heckerman, C. Kadie // Proc. 14th Conf. Uncertainty in Artificial Intelligence. – 1998 – P. 43-52.
15. B. Sarwar Item-Based Collaborative Filtering Recommendation Algorithms / B. Sarwar, G. Karypis, J. Konstan, J. Riedl // Proc. 10th Int’l WWW Conf. – 2001 – P. 285-295.
16. J. Delgado Memory-Based Weighted-Majority Prediction for Recommender Systems / J. Delgado, N. Ishii // Proc. ACM SIGIR ’99 Workshop Recommender Systems: Algorithms and Evaluation. – 1999 – P. 186-198.
17. A. Nakamura Collaborative Filtering Using Weighted Majority Prediction Algorithms / A. Nakamura, N. Abe // Proc. 15th Int’l Conf. Machine Learning. – 1998 – P. 395-403.
18. I. Soboroff Combining Content and Collaboration in Text Filtering / I. Soboroff, C. Nicholas // Proc. Int’l Joint Conf. Artificial Intelligence Workshop: Machine Learning for Information Filtering. – 1999 – P. 86-91.
19. M. Claypool Combining Content-Based and Collaborative Filters in an Online Newspaper / M. Claypool, A. Gokhale, T. Miranda, P. Murnikov, D. Netes, M. Sartin // Proc. ACM SIGIR ’99 Workshop Recommender Systems: Algorithms and Evaluation, Aug. – 1999.
20. D. Billsus User Modeling for Adaptive News Access / D. Billsus, M. Pazzani // User Modeling and User-Adapted Interaction. – 2000. – Vol. 10, No2, No3. – P. 147-180.
21. A.I. Schein Methods and Metrics for Cold-Start Recommendations / A.I. Schein, A. Popescul, L.H. Ungar, D.M. Pennock // Proc. 25th Ann. Int’l ACM SIGIR Conf. – 2002 – P. 253-260.
22. А. В. Заболеева-Зотова Латентный семантический анализ: новые решения в Internet / А. В. Заболеева-Зотова, А. Ю. Пастухов, П. В. Сердюков, Н. А. Козлова, С. А. Чернов // Информационные технологии. – 2001 – P. 67-82.

23. M. Pazzani A Framework for Collaborative, Content-Based, and Demographic Filtering // Artificial Intelligence Rev. – 1999. – P. 393-408.
24. M. Balabanovic Fab: Content-Based, Collaborative Recommendation / M. Balabanovic, Y. Shoham // Comm. ACM. – 1997. – Vol. 40, No3. – P.66-72.
25. P. Melville Content-Boosted Collaborative Filtering for Improved Recommendations / P. Melville, R.J. Mooney, R. Nagarajan // Proc. 18th Nat'l Conf. Artificial Intelligence. – 2002 – P. 187–192.
26. E. Aïmeur, G. Brassard, J. M. Fernandez, and F. S. M. Onana, Privacy-preserving demographic filtering, in Proceedings of the ACM symposium on Applied computing. New York, NY, USA: ACM, 2006 – P. 872–878.
27. Powers, David M. W. Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation // Journal of Machine Learning Technologies, 2011 – P. 37–63.
28. Ruby [Електронний ресурс] // Режим доступу: <https://www.ruby-lang.org/en/>
29. Ruby On Rails [Електронний ресурс] // Режим доступу: <http://rubyonrails.org/>
30. HTML [Електронний ресурс] // Режим доступу: <http://htmlbook.ru/html>
31. CSS [Електронний ресурс] // Режим доступу: <https://htmlbook.ru/css>
32. JavaScript [Електронний ресурс] // Режим доступу: <https://learn.javascript.ru/>
33. PostgreSQL [Електронний ресурс] // Режим доступу: <http://www.postgresql.org/>
34. Google Chrome [Електронний ресурс] // Режим доступу: <https://www.google.com/chrome/>
35. Когулько О.С. Використання методів колаборативної фільтрації для роботи рекомендаційної системи / Міжнародна науково-практична конференція «Математичне та імітаційне моделювання систем» (МОДС-2018) – м. Київ., 25-29 червня 2018 р. – С. 83-86.
36. Когулько О. С., Попенко В.Д. Надання рекомендацій елементів на основі гібридної фільтрації/ О.С. Когулько, В.Д. Попенко / Всеукраїнська науково-практична конференція молодих вчених та студентів «Інформаційні системи та технології управління» (ІСТУ-2018) – м. Київ.: НТУУ «КПІ ім. Ігоря Сікорського», 29-30 листопада 2018 р – С. 61-65.

ДОДАТОК А Графічний матеріал

Плакат 1. Діаграма варіантів використання

Плакат 2. ER-діаграма

Плакат 3. Схеми роботи алгоритмів

Плакат 4. Діаграма класів

Плакат 5. Діаграма послідовності

Плакат 6. Діаграма компонентів

Плакат 7. Діаграма розгортання